NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

TOWARD LARGE-GRAPH COMPARISON MEASURES TO UNDERSTAND INTERNET TOPOLOGY DYNAMICS

by

Lee Hsu Ann Daryl

September 2013

Thesis Advisor:	Ralucca Gera
Second Reader:	Robert Beverly

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
24–9–2013	Master's Thesis	2012-01-01—2013-09-28

4. TITLE AND SUBTITLE

TOWARD LARGE-GRAPH COMPARISON MEASURES TO UNDERSTAND INTERNET TOPOLOGY DYNAMICS

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Lee Hsu Ann Daryl

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Singapore Technologies Electronics (Info-Software Systems)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited

13. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.IRB Protocol Number: XXXX

14. ABSTRACT

By measuring network changes, we can get a better understanding of a network. Extending this to the Internet, we are able to understand the constantly occuring changes on an international scale. In this research, we propose a measure that conveys the relative magnitude of the change between two networks (i.e., Internet topology). The measure is normalised and intuitively gives an indication of whether the change is small or large. We start off by applying this measure to standard common graphs, as well as random graphs. These graphs were first simulated and the measurements taken; results were then proved theoretically. These corresponded to the simulation results, thus demonstrating correctness. For case studies, we compared actual implemented networks with that which is inferred by probes. This comparison was done to study how accurate the probes were in discovering actual network topology. Finally, we conducted real-world experiments by applying the measurements to certain segments of the Internet. We observed that the measurements indeed do pick up events which significantly influenced structural changes to the Internet.

15. SUBJECT TERMS

Distance, Dissimilarity between graphs, Symmetric difference, Internet Topology, Egypt/Libya revolution

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	UU	89	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

TOWARD LARGE-GRAPH COMPARISON MEASURES TO UNDERSTAND INTERNET TOPOLOGY DYNAMICS

Lee Hsu Ann Daryl, Civilian, Singapore Technologies Electronics (Info-Software Systems)
B.Eng., National University of Singapore, 2001

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN APPLIED MATHEMATICS

from the

NAVAL POSTGRADUATE SCHOOL
September 2013

Author: Lee Hsu Ann Daryl

Approved by: Ralucca Gera
Thesis Advisor

Robert Beverly
Second Reader

Carlos Borges
Chair, Department of Applied Mathematics

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

By measuring network changes, we can get a better understanding of a network. Extending this to the Internet, we are able to understand the constantly occuring changes on an international scale. In this research, we propose a measure that conveys the relative magnitude of the change between two networks (i.e., Internet topology). The measure is normalised and intuitively gives an indication of whether the change is small or large. We start off by applying this measure to standard common graphs, as well as random graphs. These graphs were first simulated and the measurements taken; results were then proved theoretically. These corresponded to the simulation results, thus demonstrating correctness. For case studies, we compared actual implemented networks with that which is inferred by probes. This comparison was done to study how accurate the probes were in discovering actual network topology. Finally, we conducted real-world experiments by applying the measurements to certain segments of the Internet. We observed that the measurements indeed do pick up events which significantly influenced structural changes to the Internet.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

Ark	Archipelago
ARPANet	Advanced Research Projects Agency Network
AS	Autonomous System
ASN	Autonomous System Number
BGP	Border Gateway Protocol
CAIDA	Cooperative Association of Internet Data Analysis
CDN	Content Delivery Network
GB	Gigabyte
GED	Graph Edit Distance
IP	Internet Protocol
IPv4	Internet Protocol version 4
ISP	Internet Service Provider
NPS	Naval Postgraduate School
NTC	Network Topology Capture
RIR	Regional Internet Registry
RTT	Round-trip time
SVD	Singular Value Decomposition
TB	Terabyte

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

Since the inception of the Advanced Research Projects Agency Network (ARPANet), whose offspring is the Internet, network measurement has been recognized as an important task as measurement would be critical for future development, evolution and deployment planning.In this research, we propose measures that intuitively indicate the magnitude of the change between two networks (i.e., Internet topologies). These measures are both computationally fast and scalable, allowing analysis of Internet-size networks. The measures that this thesis introduces effectively capture both the structure and the change of the network in time. By applying these measures to the Internet, we get to observe possible indicators of events that influence structural changes on a portion of the Internet. We are also able to compare the accuracy of network topology discovered by probes versus the actual implemented topology. Our tests on datasets of three consecutive years for Egypt and Libya show that the measure picks up the disruptions to Internet communications during the Arab Spring revolution in the first months of 2011. During this period of disruption, the measure exceeded 40%, as compared to regular times (i.e., periods when there was no civil unrest) when it was below 5%.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

This study would not have been possible without the help of many people.

First and foremost, I would like to express my sincerest gratitude to my advisor Dr. Ralucca Gera, whose selfless time and care were a great motivation for me. Her enthusiasm, patience and insight are greatly appreciated. I am very grateful to Dr. Robert Beverly, without whose acumen, knowledge and zest this study would not have been successful. Special thanks also go to Dr. Geoffrey Xie and Dr. Justin Rohrer for their suggestions and guidance.

I would like to thank my sponsor, Singapore Technologies Electronics (Info-Software Systems), for this rewarding opportunity at the Naval Postgraduate School.

Last, but not least, to my family and loved ones for their understanding and encouragement; especially my wife, Ching Ee, and children Rachel, Ariel and Rae-anne, for their constant support and unconditional love.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

The advent of the Internet has brought about the unprecedented revolution in the world of computers and communications. Today, the Internet provides the means for individuals and their computers to interact without physical boundaries, regardless of geographic location.

From a mathematical standpoint, topology studies the basic properties of space and its behavior under continuous deformations such as stretching and bending. For our research, we extend this study into the realm of the Internet by concerning ourselves with the connectivity changes between different computers.

The Internet is comprised of routers, where each router has multiple interfaces that connect to other routers. As traffic arrives, routers perform a lookup operation in order to forward the traffic along to the next router closer to the final destination. Thus, "connectivity" in this thesis concerns the connections between these interfaces, but not necessarily the physical wires that connect these routers. This connectivity connects companies in cooperative competition, and thus, routing decisions are influenced by economics. Users do not have the ability to influence the path that their packets take. As such, the portion of the visible, logical Internet topology is generally dictated by packet routing, which is in turn economically driven.

We study this topology by abstracting the Internet into a graph, $G(E,V)$, where E represents the set of edges and V the set of vertices. The set of vertices represent the interfaces of the routers; and edges represent the logical inter-connection between these interfaces.

1.1 Why Measure the Internet?

Since the inception of the ARPANet, whose offspring is the Internet, network measurement has been recognized as an important task as: measurement would be critical for future development, evolution and deployment planning [1].

Now, myriad people and organizations have different reasons for measuring the Internet. These interests are usually commercial, social or technical in nature.

Commercially, the ability to sell or provide information about a product to a large number of people requires a variety of Internet measurements. Such information could include the

reach of the Internet and the number of households with connectivity in a given area. Some measurements are so important that companies have been created to provide such information. In fact, mapping the Internet is the core business of Content Delivery Network (CDN) operators such as Akamai [2]. A CDN is an infrastructure for efficient delivery of Web-related content to Internet users, and it consists of a large distributed system of servers that are deployed across the Internet. According to [3], Akamai alone serves 15 to 20% of all Web traffic. A CDN service is distinguished by its ability to supply on-demand capacity to content providers, and its ability to improve performance and due to its physical proximity to the user, improve the user's access to content. These factors that distinguish a CDN obviously depend on Internet measurements.

Socially, such measurements can provide insights into popular issues. Stemming from its widespread use, governments and research institutions may also desire such information to study social implications resulting from Internet use. From [4], we observe that Internet communications were purposely disrupted in response to civilian protest and threats of civil war. An Internet blackout was imposed by the government who deemed that other forms of censorship during Arab Spring were either impractical or ineffective. Another study [5] assessed that many authoritarian states, concerned with the power of new technologies to catalyze political change, have taken various measures to filter, monitor or otherwise obstruct free speech online. Thus, governments have been known to use selective censorship to restrict Internet traffic for political, religious and cultural reasons.

Technically, Internet measurements help us to understand and influence the design of network components and protocols that drive the very nature of the Internet. Understanding the network topology helps in identifying areas with possible performance problems and provides insight into how applications can adapt.

1.2 Why Is the Internet Hard To Measure?

In [6], Floyd et al. point out that the Internet is difficult to measure due to its scale, heterogeneity, and dynamics. The Internet is like an enormous living organism. Measuring the Internet is also problematic in that the Internet is itself changing. The number of components in the Internet is constantly growing due to its design for performance, redundancy and availability of subnetworks. Its complexity and dynamism are such that it is impossible to guarantee that packets sent one after another would take the exact same path. Regardless of the quality of measures obtained, these measures may not apply elsewhere or at another point in time.

Technical, social and political factors also inhibit our ability to obtain measurements. Internet devices do not always provide appropriate measurements useful for understanding the network. Useful measures are made inaccessible due to the architecture of the Internet itself. For example, routing changes can increase delay or decrease available bandwidth; route flapping can cause packet reordering; and overloaded links or deficient router queue implementations can cause packet loss due to congestion [7]. Collection of information on the Internet usually results in datasets that are difficult to store, transfer, process and analyze. For example, the Cooperative Association of Internet Data Analysis (CAIDA) dataset [8] used in our case studies, as collected by one team averages $\approx$4.4 Gigabyte (GB) for each cycle of probing. There are about 165 cycles of probing for a year and three teams that conduct the probing. All in all, a year's worth of probing would result in a dataset that amounts to about 2.2 Terabyte (TB). To exacerbate matters, converting this dataset originally in binary format into textual form for analysis would increase its size even further.

Commercial service providers often retain proprietary information and do not share details of their networks, as the network's configuration and topology can contain personal and business data that can violate privacy and raise security concerns. The introduction of the World Wide Web and the privatization of the Internet left no framework for adequate tracking and monitoring of the Internet. Internet Service Providers (ISPs) are profit driven and are more interested in meeting the demands of their rapidly growing customer base, than in gathering and analyzing network performance data.

1.3 Research Question

The Internet is an interconnected system of networks that connects computers around the world. How can we measure the extent of these changes in interconnectivity for such a complex and large-scale network?

Given the size of the Internet, the measure needs to be intuitive, fast to compute and scaleable (i.e., capable of comparing networks of various sizes). By intuitive, we mean that the computation of the measure captures the change in the network, and the result from the measure indicates if the change was small or large. Thus, the measure's result would be a scale.

Datasets of information collected on the Internet can be used as a ready source of data. The measure can then be applied on these data to give intuitive results.

1.4 Results

We proposed innovative measures and applied it in several ways. We first compared against ground truth, which was obtained from a configuration file of a large university network as detailed in Section 5.3, with the same topology as inferred from CAIDA's datasets. This comparison against ground-truth was performed to find the difference between the actual topology and that which was discovered by probes. We then applied the measures to specific portions of the Internet as a case study, from which we identified events that influenced Internet connectivity changes in Egypt and Libya during the Arab Spring. This is an indication that our measures could pick up signals of social events.

1.5 Organization of Thesis

In investigating the research question, this thesis is organized as follows:

- Chapter 1 discusses the motivation of the research.
- Chapter 2 discusses prior and related work in the fields of measuring the Internet.
- Chapter 3 introduces the machinery used in the analysis conducted in the course of the research.
- Chapter 4 details the data used and the methodology that was taken.
- Chapter 5 contains the results of case studies conducted using our measures.
- Chapter 6 contains the summary and discusses possible areas for future work.

CHAPTER 2:
Background

In this chapter, we introduce the reader to Internet topology. We take a look at existing measures and also put forward our measurements. By measurements, we mean performing analysis on data that has been collected on the Internet and producing a reading as a result.

2.1 Overview of the Internet

The Internet is an extremely large and complex network. The Internet works on the Internet protocol suite, which consists of a networking model and a set of communications protocols. The addressing scheme for this protocol suite is the Internet Protocol (IP) address. Routing is the process by which two nodes in the network find a path to each other.

The Internet can be imagined as a collection of individual "communities" called Autonomous Systems (ASes), such as large companies and ISPs, and the physical infrastructure that connects ASes is referred to as backbones. The backbones connect ASes, and thus, ASes that are connected (i.e., adjacent), can communicate with each other. An AS is a network or a group of networks under a single administrative domain. ASes have a unique routing policy for their networks. Each AS is identified by a unique 32-bit number. Treating traffic flowing within an AS as internal, ASes help to draw the boundaries between external and internal routing. Routers (e.g., R11, R12, R21 and R31 in Figure 2.1) are devices that forward data between networks. A router is connected, via its interfaces, to two or more data lines from different networks. Each router builds up its routing table, which is essentially a list of preferred routes between any two destination addresses on interconnected networks. With multiple routers on a network, these routers can exchange information about destination addresses using routing protocols. Internal routing protocols are responsible for routing inside ASes, and likewise, external routing protocols govern routing outside of ASes. External routers see ASes as a group of a few large networks. Internal subnetting (i.e., dividing a network into two or more networks) within an AS is transparent to external routers, and external routing information is not affected by internal routing information. As such, an AS, say a large company, can subdivide its networks according to its needs, without affecting its external routing rules. Thus ASes help to reduce the load of routing by reducing the number of entries of networks in its routing table and by delegating part of the responsibility of routing internally to the company possessing that network.

A simplified representation of the Internet is shown in Figure 2.1 where the ASes are identified by their respective Autonomous System Numbers (ASNs). In this illustration, backbones connect ASN 1 to ASNs 2 and 3.

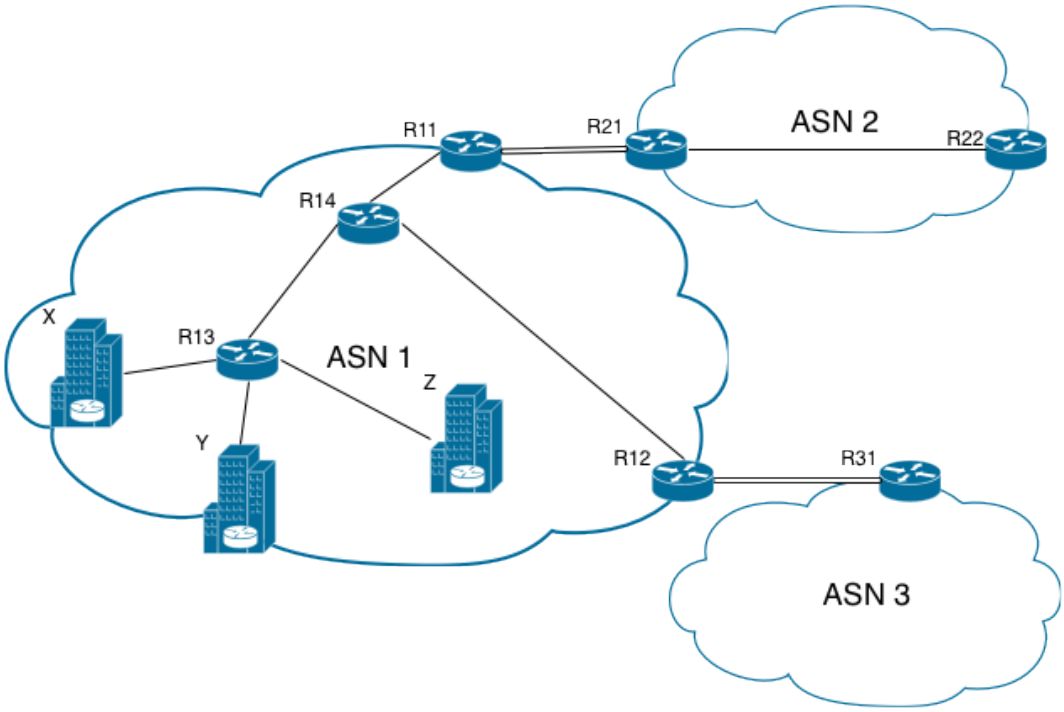


Figure 2.1: The Internet simplified.

2.2 Internet Topology

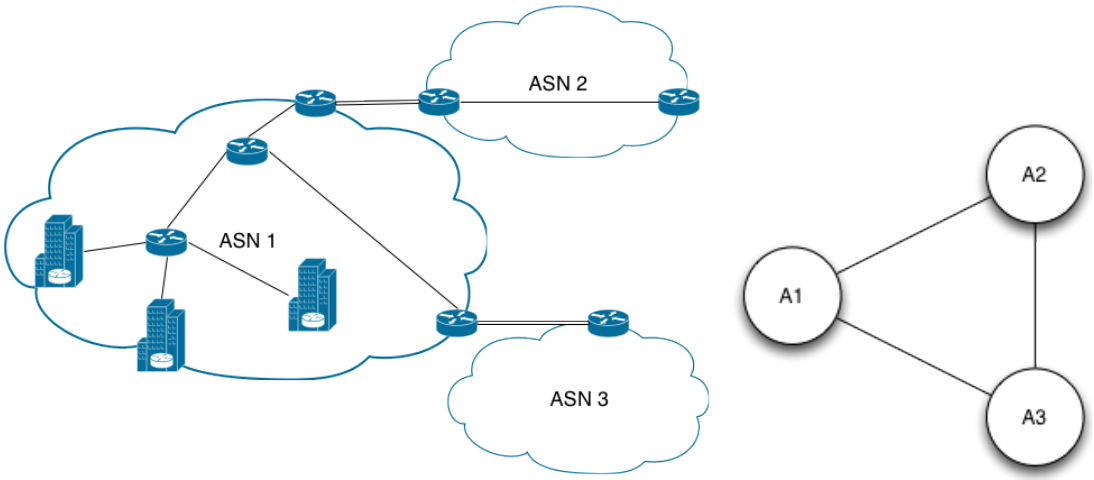
Internet topology deals with finding the topological structure of the Internet. It is daunting to map the entire hierarchy of the Internet due to its sheer size and also the rate at which the Internet is growing and evolving. The effort to map the Internet is usually incomplete and out of date the moment it appears.

2.2.1 Levels of Internet Topology

There are different granularities at which we can study the topology of the Internet as we detail in this section. For each level of granularity, we represent the network according to how its components are connected and its corresponding graph representation(s). Among these levels of granularity mentioned, interface-level mapping offers the greatest amount of information with regard to connectivity. Our research limits itself to interface-level mapping, as detailed in

the next paragraph, and the connectivity of certain segments of the Internet is what we will be investigating in the following chapters.

AS-level topology. At the AS level, routers and subnets of a single AS are represented as one single node; and the adjacency relations between the ASes are represented by edges between the nodes. These relations can be generalized into provider-customer and peer relations [9]. A customer pays its provider for connectivity to the Internet outside of its administrative domain. A pair of peers agrees to exchange traffic between their respective customers free of charge. Thus, interdomain routing policies are constrained by contractual commercial agreements between administrative domains. The AS-level representation of Figure 2.1 is shown in Figure 2.2.

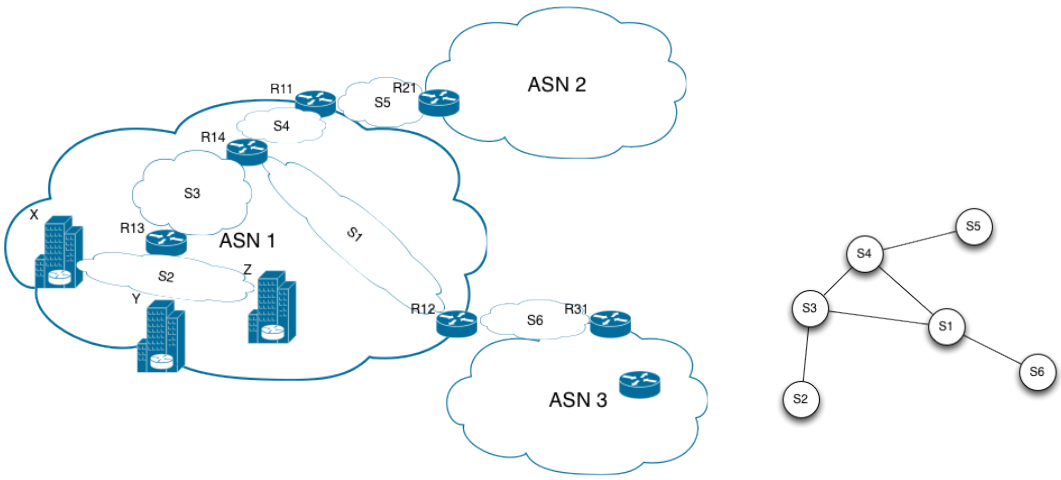


(a) Network map. (b) Graph.

Figure 2.2: AS-level representations.

Subnet-level topology. Subnet-level mapping [10] involves discovering the IP addresses that are hosted on the same subnet. A subnet is defined by the set of interfaces that it accommodates. Representing a subnet as a vertex in a graph as shown in Figure 2.3b, a link between two subnets would represent the router that directly connects these two subnets to each other.

Interface-level topology. Interface-level mapping expands on subnet-level mapping in that interfaces of routers and end hosts are described as well. Here, an interface is represented by a node and each direct connection between a pair of nodes is represented by a link or an edge. Recall that a router has more than one interface, so more than one graph might be required to capture these interface connections between routers. Figure 2.4 illustrates some of these graphs as seen from the perspective of different end points.



(a) Network map. (b) Graph.

Figure 2.3: Subnet-level representations.

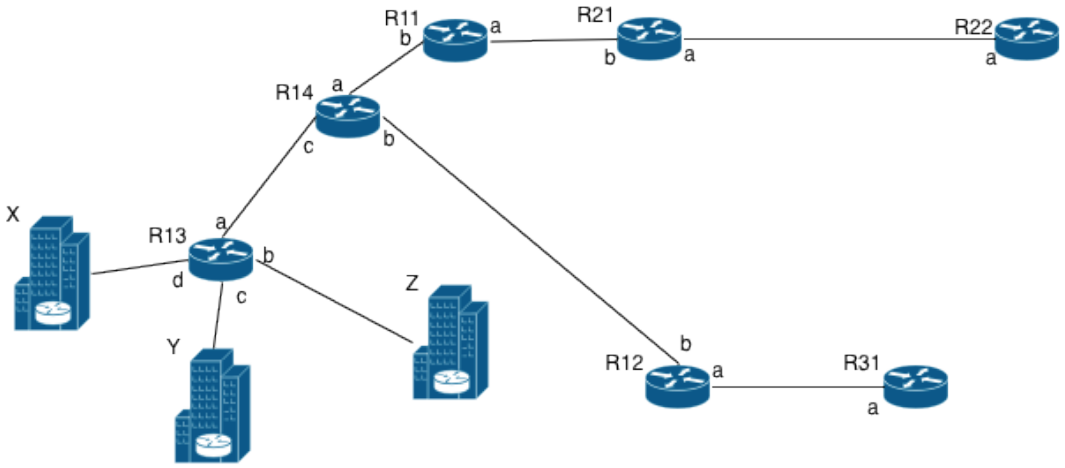
Router-level topology. Since a router can have more than one interface, each of which can have a different IP address, these interfaces within the same router can be determined by IP Alias Resolution[2] and grouped as one node. Thus, connections between node-pairs, or equivalent router-pairs, are represented by the links. Topology that is inferred from these connection-maps more closely resembles what is on the ground as the devices are physically routed in this manner. However, the information required for IP Alias Resolution is not easily obtainable, and techniques to do so are still under research [11–14].

2.2.2 Obtaining Network Topology

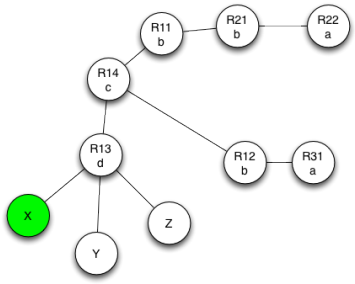
There has been much research in the area of Network Topology Capture (NTC) algorithms. In [15], adaptive probing techniques were proposed. These techniques leveraged external knowledge and data from prior cycle(s) to guide the selection of probed destinations and assignment of destinations to vantage points. By exploiting structural knowledge of the network, measurement cost was reduced. Many NTCs aim to capture network dynamics with a reduced discovery budget (i.e., number of probes), while maintaining a broad coverage of the network topology at the same time [16].

However, the results obtained from NTCs remain incomplete or incorrect due to inherent limitations in the collection. Many systems may be offline at the time when the probing was performed. Also, network obstacles, such as firewalls and load balancers, often prevent probes

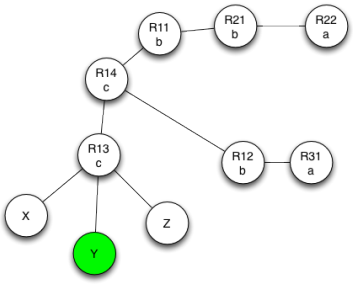
[2]The process of identifying IP addresses belonging to the same router.



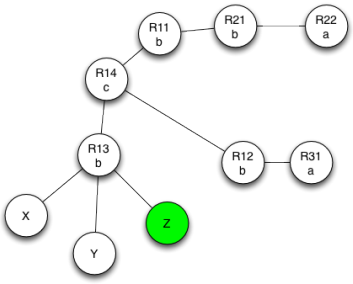
(a) Network map.



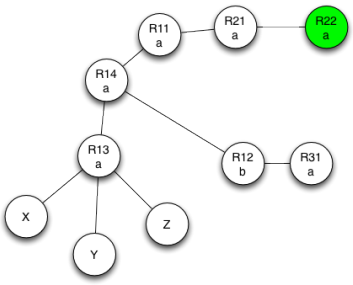
(b) Graph of interfaces as seen from X.



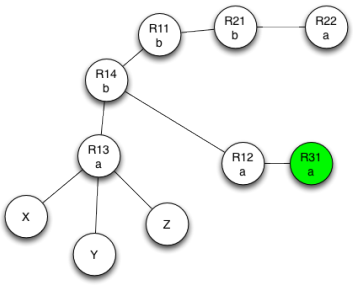
(c) Graph of interfaces as seen from Y.



(d) Graph of interfaces as seen from Z.



(e) Graph of interfaces as seen from R22.

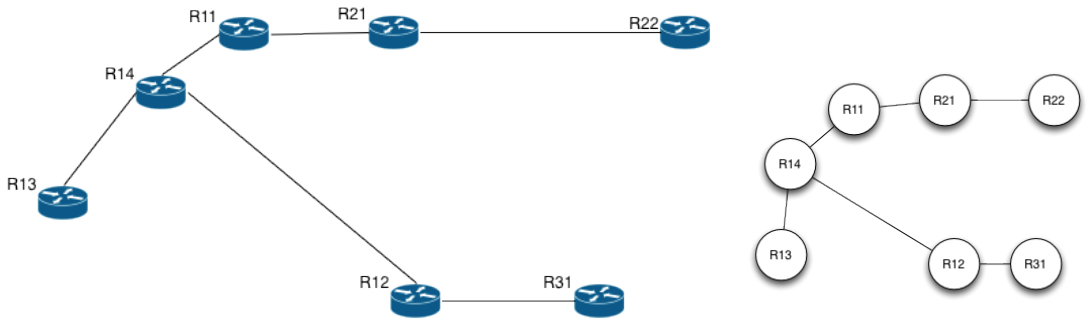


(f) Graph of nterfaces as seen from R31.

Figure 2.4: Interface-level representations.

from reaching their target destinations or may route the probes in an unexpected way.

Ground-truth, which we refer to as the actual network implementation, would prove to be an ideal source of information for comparing the accuracy of results obtained from probes. Unfor-



(a) Network map. (b) Graph.

Figure 2.5: Router-level representations.

tunately, an organization's network layout is a critical piece of information as it usually reveals the structure of the main means of communication in the organization. If this information were to fall into the wrong hands, an organization would allow itself to be susceptible to sabotage. Thus, due to security concerns, ground-truth information is not openly available.

Nonetheless, despite the aforementioned challenges, the importance of measuring the Internet and its "demand" for data to work with has prompted some organizations to curate their results as data for use by the research community. For example, as part of its objective to collect and share data for scientific analysis of the Internet, CAIDA has a collection of datasets [17] resulting from both active and passive measurement of the Internet. We will be using some of these datasets in our research.

2.2.3 Traceroute

Traceroute [18] is a computer network tool used to show the route taken by packets across an IP network. The history of the route is recorded as the Round-trip times (RTTs)[3] of the packets received from each successive host in the route.

Paris traceroute [19, 20] is an improved version of the original traceroute program. The original traceroute did not perform well in the presence of routers that employ load balancing on packet header fields, since traceroute discovers hops along a route with a series of probe packets, while a load-balancing router can direct these probes along different paths. This led to inaccurate and incomplete paths, which resulted in erroneous network maps. By controlling packet header contents, Paris traceroute obtains a more precise picture of the actual routes that packets follow.

[3]The length of time it takes for a data packet to be sent plus the length of time it takes for an acknowledgement of that data packet to be received.

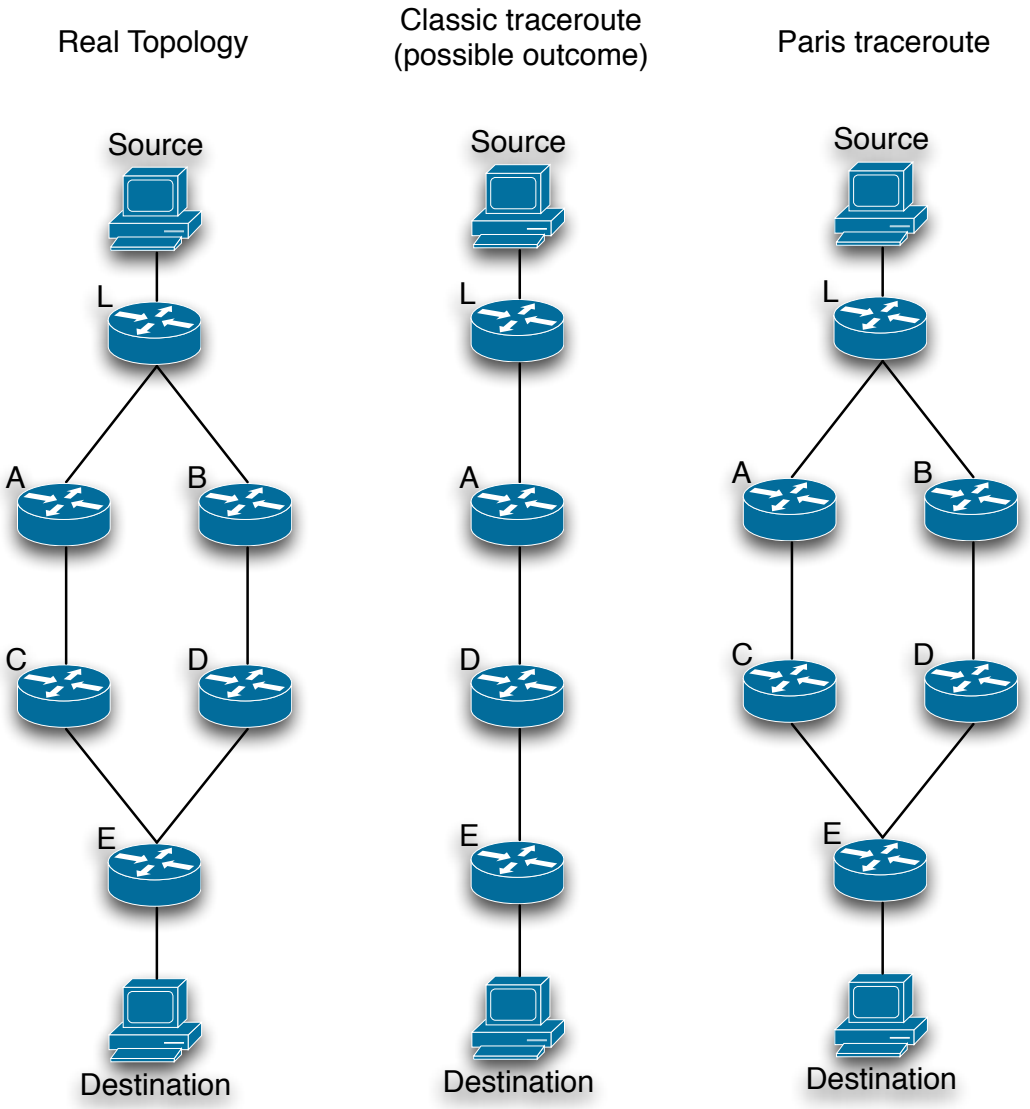


Figure 2.6: Classic versus Paris traceroute adapted from [19].

Figure 2.6 illustrates how the Paris traceroute is preferred over the classic traceroute. Suppose we wanted to measure the route from a source to a specified destination. The ground truth is shown on the left, where router L balances load across across two paths, via routers A or B. The middle diagram in the figure shows a possible outcome when the classic traceroute is used. On

the right is the output obtained by using the Paris traceroute, which more closely resembles the real topology.

2.3 Existing Measurements

Network topology ASes will be modeled by graphs in order to facilitate measurements. The interface-level maps will be represented by graphs, with vertices denoting the interfaces and edges denoting the pair-wise connection between the interfaces. (We detail our approach in Section 4.2.)

There have been several ways to measure graphs. Most of the existing ones use statistical results about the Internet. In the following sub-sections, we briefly mention some standard ways of measuring networks by representing them as graphs. We will take a different approach in this thesis. (A formal introduction to Graph Theory is in Section 3.1.1.)

2.3.1 Primitives

These are some of the basic forms of measurements typically used in graph theory which serve as the foundation of more elaborate ones that follow. These measures give information about the graph but do not compare graphs, as we wish to do for our research. We do not compare our measures to these primitives. Nonetheless, these primitives are included here so that the reader is aware of innate measurements from graph theory. The terms and definitions used here are those found in [21].

Distance. If a graph G has a u,v-path, then the distance from u to v, written $d_G(u,v)$ or simply $d(u,v)$, is the least length (or number of edges) of a u,v-path. lf G has no such path, then $d(u,v) = \infty$.

Diameter. The diameter of a graph G, denoted *diam G*, is the maximum distance between any two vertices in the graph, i.e., $diam\ G = max_{u,v \in V(G)} d(u,v)$.

Eccentricity. The eccentricity of a vertex u, written $\varepsilon(u)$, is the maximum distance of the specified vertex to all vertices in the graph, i.e., $\varepsilon(u) = max_{v \in V(G)} d(u,v)$. Informally, this is the largest distance between any two vertices in the graph.

Radius. The radius of a graph G, written *rad G*, is the minimum eccentricity of all vertices in the graph. i.e., $rad\ G = min_{u \in V(G)} \varepsilon(u)$. Informally, this is the smallest distance between any two vertices in the graph.

2.3.2 Statistical Analysis

Statistical analysis of network structure usually pertains to the selective pairing of identified types of vertices, also known as assortative mixing. The converse is referred to as disassortative mixing. Maslov et al. proposed in [22] that the structure of the Internet is characterized by three identified types of vertices: providers, who run the Internet backbone having high levels of connectivity; consumers (i.e., end users) of the Internet service; and the ISPs who join the providers and consumers. This particular form of assortative mixing can be seen as selective linking according to a scalar vertex property. In this case, the mixing is according to vertex degree and is commonly known as degree correlation. Several ways of quantifying degree correlations have been proposed in previous research. In [23, 24], studies of the Internet calculated the mean degree α of the network neighbors of a vertex as a function of the degree k of that vertex. A network is deemed to be assortatively mixed if α increases with k, and disassortative if otherwise. The latter was found to be the case for the Internet.

2.3.3 Centrality

There are various types of measures for the centrality of a vertex within a graph. These determine the relative importance of a vertex within the graph. Some of those that are widely used are mentioned here.

Degree centrality, which is defined as the number of edges incident upon a vertex. Note that this is the same as the degree defined above, and some researchers use it as centrality. This can be further decomposed to indegree and outdegree which differentiates between the edges directed towards or out of the vertex in question respectively.

Betweenness, when applied to vertex, quantifies the number of times a node acts as a bridge along the shortest path between two other vertices. Similarly, when applied to edges, it quantifies the number of times an edge is traversed along the shortest path between two other vertices.

Closeness is defined as the inverse of the sum of distances from a vertex to all other vertices using the shortest path. The more central a vertex is, the lower is its total distance to all other vertices, and thus a higher closeness centrality.

Eigenvector centrality works by assigning relative scores to all vertices in the network based on the concept that connections to high-scoring vertices contribute more to the score of the vertex in question than equal connections to low-scoring vertices. In other words, it measures the influence that a vertex has on a network. This can be computed using Singular Value Decom-

position (SVD).

2.3.4 Graph Edit Distance (GED)

The **Graph Edit Distance (GED)** [25] measures the minimum number of graph edit operations that are required to transform one graph to the other. These operations include:

- insert/delete isolated vertex
- insert/delete an edge
- change the label of a vertex/edge (if labeled graphs)

Thus, the more dissimilar two graphs are, the more transformation operations have to be performed. The time and space complexity involved in computing the GED of two graphs is exponentially proportionate to the number of vertices of the two graphs. For large graphs such as the Internet, the GED problem is computationally demanding, and in general, considered NP-Hard[4].

2.3.5 Spectral Analysis

Clustering and spatial properties of the ASes of the Internet were characterized by Gkantsidis et al. [27] using spectral analysis. The analysis captured significant information about the clustering properties of the topology. Spectral analysis examines the eigenvectors corresponding to the largest eigenvalues of the normalized transposed adjacency matrix of the topology in question. The output is a plot which consists of a specified number of the largest eigenvalues which loosely corresponds to the eigenvectors of the main clusters in the topology.

2.4 Thesis Contribution

The existing primitive and statistical measures apply mostly to measurements taken within the graph. The GED and spectral analysis have been applied between different graphs and also across time. However, they give results that might require a further level of interpretation.

The GED gives a number that quantifies the number of operations required to get from one graph to another, and this does not reflect the magnitude of the relative change between the two graphs, which is what interests us. From the GED, one was not able to determine how different the two graphs in comparison were, unless it was applied to two graphs with exactly the same

[4] A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem. NP-hard therefore means "at least as hard as any NP-problem" although it might, in fact, be harder. [26]

vertex set [28] which is not feasible in our case. Several samples might have to be taken to determine if the change was comparatively small or large.

The spectral analysis gave a range of values which justly describes the different characteristics captured by the analysis.

Our proposed measures capture the difference between two networks as a single number or index. This index intuitively indicates the magnitude of the change on a normalized scale. Thus, without making further comparisons, the measure indicates the "size" of the change.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3:
Behind the Scene

In this chapter, we bring forth mathematical concepts that might help the reader view the world of networks in a mathematical perspective. Here, we also introduce *quasi semi metrics* which shed light when comparing graphs. We will also touch on these measures' performance and scalability when applied to large networks such as the Internet. These measures are chosen primarily due to their applicability, as well as, how easy it is to understand their computation and the meaning of the result, their scalability and speed for computation.

3.1 Preliminaries

Each of our proposed measures gives a simple index (percentage of change). From the indices, it is immediately intuited by the reader how much the two graphs have changed in terms of vertices or edges. To derive at these indices, we first translate raw collected data into graphs using basic graph theory. After which, set operations are performed on either the vertex set or the edge set of the graphs. With the massive amount of data, appropriate visualization methods are used as they complement our measures. Much of the research would not have made much progress without the aid of these visualizations.

3.1.1 Graph Theory Terminology and Concepts

A majority of the terms and definitions used in this paper are those found in [21]. Terms and symbols not found in that text are referenced appropriately.

A **graph** G is a triple consisting of a vertex set $V(G)$, an edge set $E(G)$, and a relation that associates with each edge two vertices (not necessarily distinct) called its **endpoints**. **Multiple edges** are edges having the same pair of endpoints. A **simple graph** is a graph having no loops or multiple edges. A **loop** is an edge whose endpoints are equal.

In this paper, we consider only simple graphs, as the study is carried out on logical connections of network interfaces. Despite the possibility that two machines are connected physically by a common wire or link, multiple interfaces on one machine connect to interfaces on the other machine in a uniquely identifiable fashion. Thus, we do not have multiple edges. Likewise, loops which connect an interface to itself are not relevant to our study.

A simple graph can thus be specified by its vertex set and edge set, treating the edge set as one of unordered pairs of vertices and writing $e = uv$ or $e = vu$, for an edge e with endpoints u and v. When u and v are the endpoints of an edge, they are adjacent and are neighbors. This is denoted by uv, meaning"u is adjacent to v". Since the data used in our analysis was collected only when probes were sent, the information flow is bi-directional, so we restrict our graph model to only undirected edges.

Complete Graph A complete graph is a simple graph of all whose vertices are pairwise adjacent. We denote the (unlabelled) complete graph with n vertices as K_n.

Complete bipartite graph (biclique) A complete bipartite graph or biclique is a simple bipartite graph in which two vertices are adjacent if and only if they are in different partite sets. When the sets have sizes r and s, the (unlabeled) biclique is denoted $K_{r,s}$.

Path A path is a simple graph whose vertices can be ordered so that two vertices are adjacent if and only if they are consecutive in the list.

Cycle A cyclic graph is one with an equal number of vertices and edges whose vertices can be placed around a circle so that two vertices are adjacent if and only if they appear consecutively along the circle.

Erdős Rényi (ER) Random Graph In 1959, Paul Erdős and Alfréd Rényi introduced the model for generating random graphs, including one that sets an edge between each pair of vertices with equal probability, independently of the other edges. Motivated by the modeling of physical properties and by the analysis of algorithms in computer science and their applicability to the Internet, we look at random graphs to model and apply probabilistic techniques for the occurrence of events. In our context, such an event could be the existence or creation of edges between any two vertices.

There are two closely related variants of the ER random graph model.

- $G(n,M)$ model. A graph is chosen uniformly at random from the collection of all graphs which have n vertices and M edges.
- $G(n,p)$ model. A graph is constructed by connecting vertices randomly. Each edge is included in the graph with probability p independent form every other edge. Equivalently, all graphs with n vertices and M edges have equal probability of $p^M(1-p)^{\binom{n}{2}-M}$.

Symmetric Difference Conventionally, if G and H are graphs with vertex set V, then the symmetric difference $G \triangle H$ is the graph with vertex set V whose edges are all those edges appearing in exactly one of G and H. The symmetric difference in this case pertains only to the set of edges.

Also, in our context, we consider graphs G and H as having different vertex sets and look not only at edges but vertices as well. We then propose two indicators "Vertex Symmetric Difference" and "Edge Symmetric Difference", as elaborated in the next section.

3.1.2 Set Theory

From [29], a complement of a set A refers to elements not in A. The relative complement of A with respect to a set B is the set of elements in B but not in A. The relative complement of A in B is formally denoted

$$B \setminus A = \{x \in B | x \notin A\}$$

3.2 Our Measures

In this section we introduce an innovative way of comparing graphs that is intuitive and fast to compute.

3.2.1 Vertex Symmetric Difference

Here, we consider only vertices, and compare two graphs G and H having two vertex sets V(G) and V(H), respectively.

Definition 3.2.1. *For two graphs G and H, we define the* vertex symmetric difference $vsd(G,H)$ *as*

$$vsd(G,H) = \frac{|V(G) \setminus V(H)| + |V(H) \setminus V(G)|}{|V(G)| + |V(H)|}.$$

Informally, the vertex symmetric difference counts the vertices that are in one graph and not the other. This is then normalized over the total number of vertices in both the graphs, without double-counting common vertices, so that it is relative to the order of the graph.

In the case where graphs G and H have exactly the same set of vertices, $vsd(G,H) = 0$. If graphs G and H are disjoint, or have totally different vertices, $vsd(G,H) = 1$.

Thus, we see that *vsd* of any two graphs lies in the closed interval $[0,1]$. i.e., $0 \leq vsd \leq 1$.

On this scale, we are able to say intuitively if the difference between the two graphs (vertex-wise) is significant.

For example, consider the two graphs in Figure 3.1.

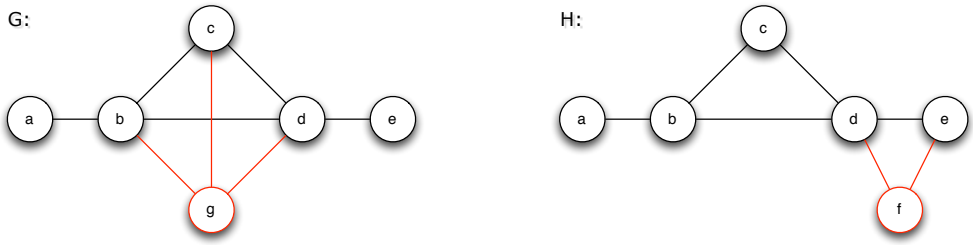


Figure 3.1: Example to illustrate *vsd* and *esd* between two graphs.

We then have

$$vsd(G,H) = \frac{|V(G) \setminus V(H)| + |V(H) \setminus V(G)|}{|V(G)| + |V(H)|} = \frac{1+1}{6+6} = \frac{2}{12} = 16.7\%$$

3.2.2 Edge Symmetric Difference

The measure introduced in this section concerns itself only with edge sets, and we call it **edge symmetric difference** and we present an instance of its use on the topology of the internet to compare snapshots of the same network from independent sources. We would like to remind the reader that all our graphs have labeled vertices, which allows us to perform symmetric difference on the set of edges of graphs.

Definition 3.2.2. *For two graphs G and H, the* edge symmetric difference $esd(G,H)$ *is defined as*

$$esd(G,H) = \frac{|E(G) \setminus E(H)| + |E(H) \setminus E(G)|}{|E(G)| + |E(H)|}.$$

Informally, the edge symmetric difference counts the edges present in one graph and not the other. This is then normalized over the total number of edges in both the graphs, without double-counting common edges, so that it is relative to the size of the graph.

In the case where graphs G and H have exactly the same edges, $esd(G,H) = 0$. If graphs G and H are disjoint, or have totally different edges, $esd(G,H) = 1$.

Thus, we see that the *esd* of any two graphs lies in the closed interval $[0,1]$. i.e., $0 \leq esd \leq 1$. On this scale, we are able to say intuitively if the difference between the two graphs (edge-wise) is significant.

For example, consider the two graphs in Figure 3.1.

We have

$$esd(G,H) = \frac{|E(G) \setminus E(H)| + |E(H) \setminus E(G)|}{|E(G)| + |E(H)|} = \frac{3+2}{8+7} = \frac{5}{15} = 33.3\%$$

3.2.3 Basic Properties of *esd*

Note that $esd(G,H)$ is not a metric since, for example, it does not satisfy triangle inequality as we show below; namely we present an example of three graphs for which $esd(G,H) + esd(H,K) = \frac{0+3}{3+6} + \frac{2+0}{6+4} = \frac{8}{15} < \frac{5}{7} = \frac{2+3}{3+4} = esd(G,K)$.

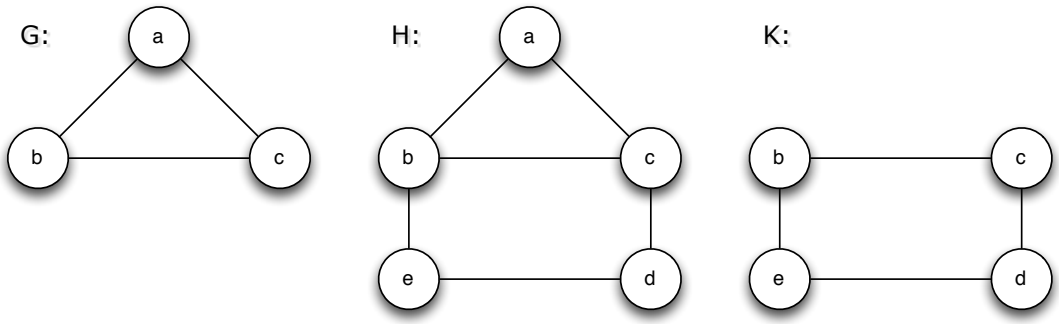


Figure 3.2: Counterexample for the triangle inequality for *esd*.

However, $esd(G,H)$ is a pseudo semimetric. Recall the definition of a pseudo semimetric.

Definition 3.2.3. *A pseudo semimetric on a set X is a function $d : X \times X \longrightarrow \mathbb{R}_{\geq 0}$ such that, for every $x, y \in X$:*

- $d(x,x) = 0$
- $d(x,y) = d(y,x)$.

By equal graphs we mean isomorphic graphs. It is easy to observe that both properties are true for the edge symmetric difference of graphs.

Proposition 3.2.4. *Let Γ be the set of all graphs. Then $esd(G,H)$ is a pseudo semimetric on Γ.*

Unlike a metric space, points in a pseudo semi-metric space need not be distinguishable; that is, one may have $d(x,y) = 0$ for distinct values $x \neq y$. And so it is interesting to study equivalence relations and equivalence classes to see what graphs we can not distinguish with this pseudometric, that is what graphs are in the same equivalence class.

Theorem 3.2.5. *Let G and H be two graphs, and define the relation $\mathscr{R}$ defined on Γ, the set of all graphs, by*

$$G\mathscr{R}H \iff esd(G,H) = 0.$$

Then $\mathscr{R}$ is an equivalence relation, whose equivalence classes are
$\mathscr{S}_G = \{H : H = G \cup \alpha K_1, \forall \alpha \in \mathbb{Z}_{\geq 0}\}$, for all graphs G without isolated vertices.

Proof: Since the reflexive and symmetric properties are easy to verify, we only show the transitivity below. Let G, H, and K be three graphs and assume that $G\mathscr{R}H$ and $H\mathscr{R}K$. This happens if and only if $\vec{e}_{G-H} = \vec{e}_{H-K} = \vec{0}$. Since $\vec{e}_{G-K} = \vec{e}_{G-H} + \vec{e}_{H-K} = \vec{0}$, it follows that $G\mathscr{R}K$. For the equivalence classes, $G\mathscr{R}H$ if and only if $\vec{e}_{G\oplus H} = \vec{0}$, i.e., $E(G) = E(H)$, and we obtain $\mathscr{S}_G = \{H : H = G \cup \alpha K_1, \forall \alpha \in \mathbb{Z}_{\geq 0}\}$ for all graphs G without isolated vertices.

This implies that if we restrict our attention to just the numerator of the $esd(G,H)$ (namely considering graphs of the same size) and we do not allow isolated vertices (which will not happen in the networks we study) then we have a metric.

Proposition 3.2.6. *Let Γ_m be the set of all graphs with m edges and no isolated vertices. Then $esd(G,H)$ is a metric on Γ_m.*

Proof: Since no isolated vertices are allowed it is easy to see now that $esd(G,H) = 0$ if and only if G, H have the same edge and vertex sets, and this happens if and only if $G \cong H$. The symmetric property is true because $\vec{e}_{G\oplus H} = \vec{e}_{H\oplus G}$. Finally, for the triangle inequality, consider this:

$$\begin{aligned} esd(G,H) + esd(H,K) &= \frac{\frac{||\vec{e}_{G-H}||^2}{2}}{2m} + \frac{\frac{||\vec{e}_{H-K}||^2}{2}}{2m} \\ &\geq \frac{\frac{||\vec{e}_{G-K}||^2}{2}}{2m} = esd(G,K). \end{aligned}$$

This completes the proof. ∎

3.2.4 *esd* for Classes of Graphs

We initially plotted the *esd* of complete and cyclic graphs as illustrated in Figure 3.3, and observed that they approach limits. This spurred us to obtain the corresponding theoretical proofs. This led to the following propositions, which initially considered other standard graphs, and is then extended to random graphs.

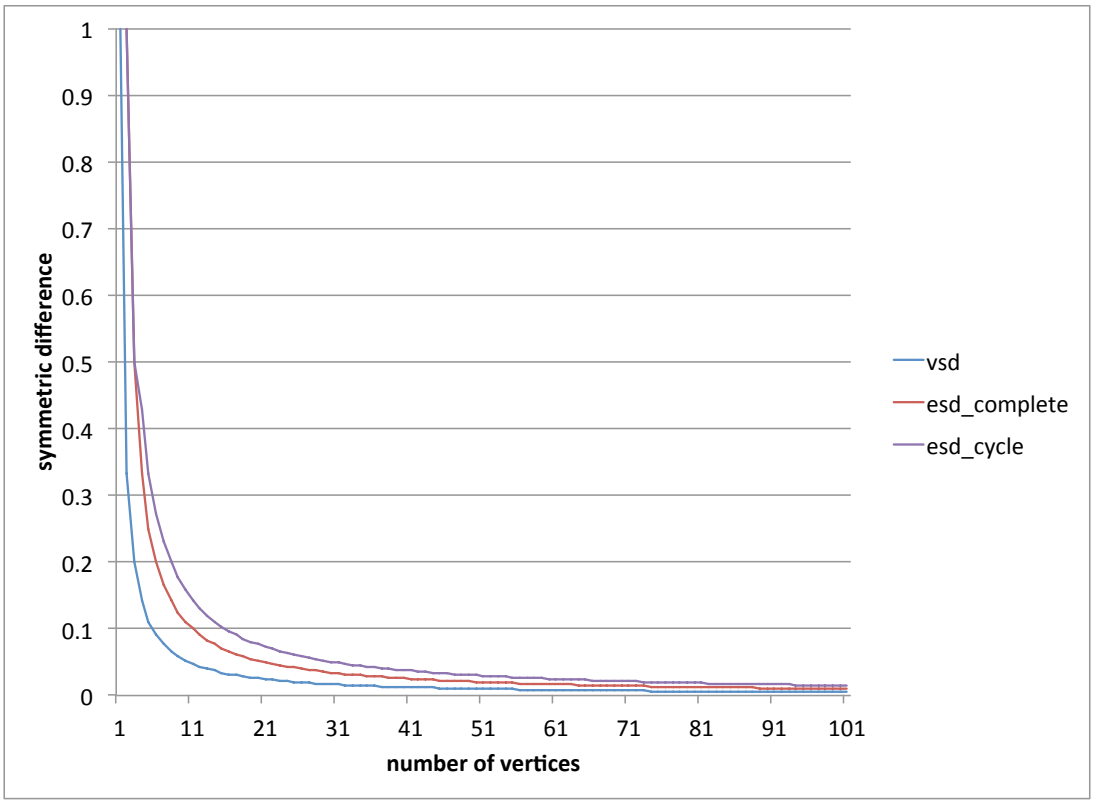


Figure 3.3: Symmetric differences for complete and cyclic graphs.

Proposition 3.2.7. *For the standard classes of graphs below, we find the edge symmetric difference.*

1. *For the complete graph* K_n $(n \geq 3)$ *we have that* $esd(K_{n+1}, K_n) = \frac{1}{n}$.
2. *For the cyclic graph* C_n $(n \geq 3)$ *we have that* $esd(C_{n+1}, C_n) = \frac{3}{2n+1}$.
3. *For the path* P_n $(n \geq 2)$ *we have that* $esd(P_{n+1}, P_n) = \frac{1}{2n-1}$.
4. *For the complete bipartite graph* $K_{n,m}$ $(n, m \geq 1)$ *we have that* $esd(K_{n+1,m}, K_{n,m}) = \frac{1}{2n+1}$.
5. *For the star graph* S_n $(n \geq 1)$ *we have that* $esd(S_{n+1}, S_n) = \frac{1}{2n-1}$.
6. *For the hypercube graph* Q_n *with* 2^n *vertices, we have that* $esd(Q_{n+1}, Q_n) = \frac{1}{3} + \frac{4}{3(3n+2)}$.

Proof: Here is the proof for each of the items above.

1. Since K_i ($i \geq 3$) has $\binom{i}{2}$ edges, it follows that

$$esd(K_{n+1}, K_n) = \frac{\binom{n+1}{2} - \binom{n}{2}}{\binom{n+1}{2} + \binom{n}{2}} = \frac{\frac{n}{2}(n+1-n+1)}{\frac{n}{2}(n+1+n-1)} = \frac{n}{n^2} = \frac{1}{n}.$$

2. C_i has $i \geq 3$ edges. Suppose the edges of C_i are $v_1v_2, v_2v_3, ..., v_{i-1}v_i, v_iv_1$. To obtain a cycle C_{i+1} we introduce a vertex v' together with two new edges $v_{i-1}v'$ and $v'v_i$, and the omission of the edge $v_{i-1}v_i$. It follows that

$$esd(C_{n+1}, C_n) = \frac{2+1}{(n+1)+n} = \frac{3}{2n+1}.$$

3. P_i has $i-1 \geq 1$ edges. Since $P_i \subset P_{i+1}$ and P_{i+1} has 1 additional edge over P_i, it follows that

$$esd(P_{n+1}, P_n) = \frac{1}{n+(n-1)} = \frac{1}{2n-1}.$$

4. $K_{i,j}$ has ij edges ($i \geq 1, j \geq 1$). Without loss of generality, say we introduce a vertex to the first partition of $K_{i,j}$ to obtain $K_{i+1,j}$. Since $K_{i,j} \subset K_{i+1,j}$ and $K_{i+1,j}$ has j additional edges over $K_{i,j}$, it follows that

$$esd(K_{i,j}, K_{i+1,j}) = \frac{j}{ij+(i+1)j} = \frac{1}{2i+1}.$$

5. Note that we define S_n here as a tree with 1 internal vertex and $n-1$ leaves. Since $S_i \subset S_{i+1}$ and S_{i+1} has an one additional edge over S_i, it follows that

$$esd(S_{i+1}, S_i) = \frac{1}{i+(i-1)} = \frac{1}{2i-1}.$$

   Alternatively, S_i is the complete bipartite graph $K_{1,i-1}$. Using part 4. above, we have,

$$esd(S_{i+1}, S_i) = esd(K_{1,i}, K_{1,i-1}) = \frac{1}{i+(i-1)} = \frac{1}{2i-1}.$$

6. Q_i has $i2^{i-1}$ edges and 2^i vertices. Since $Q_i \subset Q_{i+1}$ and Q_{i+1} has $i2^{i-1} + 2^i = (i+2)2^{i-1}$

additional edges over Q_i, it follows that

$$esd(Q_{i+1}, Q_i) = \frac{(i+2)2^{i-1}}{(i+1)2^i + i2^{i-1}} = \frac{i+2}{3i+2} = \frac{1}{3} + \frac{4}{3(3i+2)}.$$

This proves the result. ∎

We now consider large complex networks, namely the random graph types (Erdös-Rényi graphs [30]) as well as scale-free graphs (like the Barabási-Albert graphs [31]). Recall that a complex network has the following properties:

- scale free: many small degree vertices held together by a few large degree vertices
- small world: small diameter (short paths between any two vertices)
- evolution: high degree vertices emerge through addition of new vertices and the preferential attachment of these vertices
- competition: vertices with high fitness (ability to attract links) become high degree vertices
- robustness: the ability of the system to maintain its function even if many neighborhoods do not function. Resilience against random error, but fragility to attacks
- communities: locally dense neighborhoods that behave similarly.

Networks with a complex topology and unknown organizing principles often appear random; thus random-graph theory is regularly used in the study of complex networks. To obtain the random graph $G(n,p)$, we start with n vertices, and every pair of vertices is being connected with the same fixed probability p. Consequently the total number of edges is a random variable with the expectation value $EV[E(G)] = p\binom{n}{2}$. Note that when p is very small, the networks are very sparse, and as p increases the network will become more connected. Now we present an example that describes the expected value of $esd(G(n+1,p), G(n,p))$ for random graphs.

It should be noted that $esd(G,H) = 1-p$, if G and H are random graphs with the same fixed probability p. To see that it is true, recall the following lemma for expected value. We use $EV(A)$ to denote the expected value of A since $E(G)$ denotes the cardinality of the edge set of G.

Lemma 3.2.8. *Let sets A and B be arbitrarily chosen elements without repetitions from set R. Then*

$$EV[|A \cap B|] = \frac{|A| \cdot |B|}{|R|}.$$

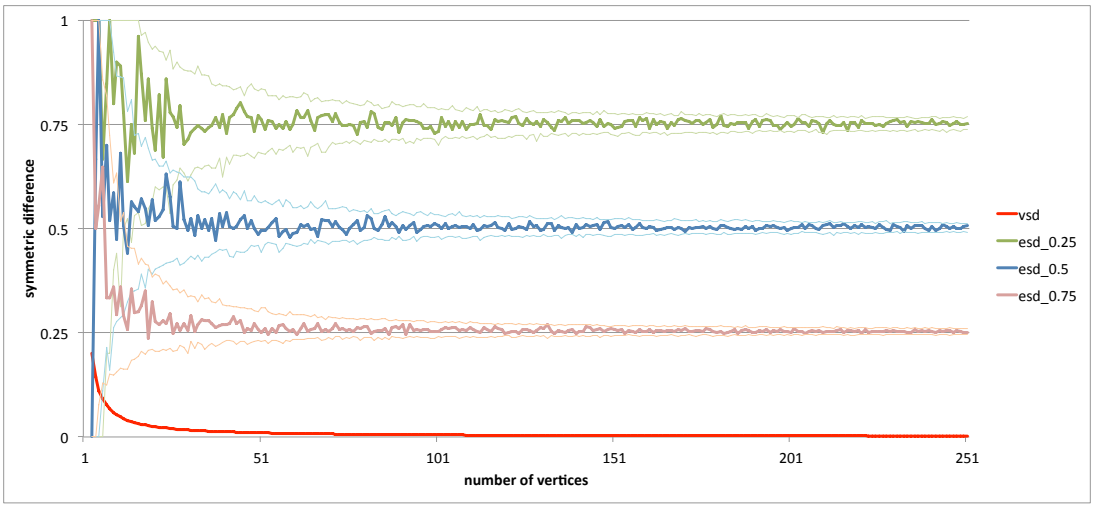


Figure 3.4: Symmetric differences for random graphs *(1000 samples; p=0.25, 0.5, 0.75)*.

Thus we have the following:

Proposition 3.2.9. *Let G and H be two random graphs of order $n+1$ and n and same fixed probability p, respectively. Then the expected value of $esd(G,H)$ approaches $1-p$, as $n \to \infty$.*

Proof: Using Lemma 3.2.8 where $A = E(G)$ and $B = E(H)$ (and $R = E(K_{n+1})$ since we are sampling from all possible edges on $n+1$ vertices, we have the following:

$$\begin{aligned}
EV[esd(G,H)] &= EV\left[\frac{|E(G)\setminus E(H)|+|E(H)\setminus E(G)|}{|E(G)|+|E(H)|}\right] \\
&= EV\left[\frac{|E(G)|+|E(H)|-2|E(H)\cap E(G)|}{|E(G)|+|E(H)|}\right] \\
&= \frac{p\cdot\binom{n+1}{2}+p\cdot\binom{n}{2}-2\frac{p\cdot\binom{n+1}{2}\cdot p\cdot\binom{n}{2}}{\binom{n+1}{2}}}{p\cdot\binom{n+1}{2}+p\cdot\binom{n}{2}} \\
&= \frac{p\cdot n^2-2p^2\cdot\frac{(n+1)n^2(n-1)}{4}\cdot\frac{2}{(n+1)n}}{p\cdot n^2} \\
&= 1-p\cdot\frac{n-1}{n}.
\end{aligned}$$

As $n \to \infty$, we have the desired result that $EV[esd(G,H)] \to 1-p$. ∎

Now, the Internet graph may look like it has a random structure at first since nobody directs the network flow or adjacency. Thus, of course, it would mean that the degree distribution would follow a Gaussian distribution, which does not happen to be the case in the snapshots that we can take of smaller networks. Modeling the Internet is still a great open problem with some breakthroughs pointing to the possibility of the Internet being a scale free network mainly due to preferential attachment that produces large hubs. Thus, research shows that the degree distribution follows a power law asymptotically x^k, where generally $2 < k < 3$ although k can follow outside of these bounds. And so, it makes sense to study this model as well. For our simulations of power law degree distributions we use the Barabási-Albert preferential attachment model [31]. The Barabási-Albert-graph (n, m, seed=None), denoted by $BA(n,m)$ returns a random graph using the Barabási-Albert preferential attachment model. That is, it returns a graph of n vertices, grown by preferentially attaching new m vertices with a fixed probability p to the high degree of preexisting vertices as follows:

- initially there are $n_0 = n \cdot p$ vertices,
- vertex $n_i + 1^{st}$ is introduced with $m = p \cdot n_i$ edges to the preexisting n_i vertices ($i \geq 0$),
- the probability, that vertex v_i will be linked to a new vertex x, is $\dfrac{\deg v_i}{\sum_{\forall v \in V(G)} \deg v} = \dfrac{\deg v_i}{2|E(G^*)|}$, where $E(G^*)$ is the set of edges in the graph before the new vertex x is attached.

Graphing the edge symmetric difference for such $BA(n, m = np)$ for $4 \leq n \leq 255$ gives the graph in Figure 3.5.

The Barabási-Albert model incorporates two important general concepts: growth and preferential attachment, which exist widely in real networks. Theoretically computing the *esd* for these graphs would be desirable. However, we leave this as an open problem since we are not able to account for preferential attachment in our theoretical computations of *esd*.

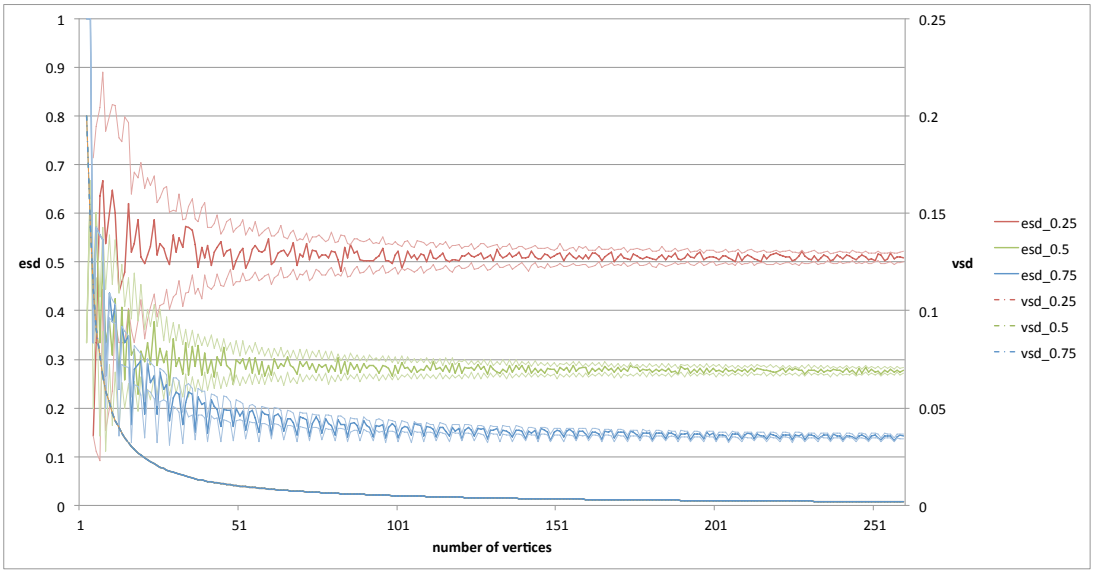


Figure 3.5: Symmetric differences for Barabási-Albert type graphs *(1000 samples; p=0.25,0.5,0.75)*.

CHAPTER 4:
Data and Methodology

In this chapter, the mathematical concepts previously discussed are applied to the real world, specifically, the Internet. Here, we analyze datasets collected on the Internet, discuss how information pertaining to the analysis is extracted and touch on the facets of analysis.

4.1 Source of Data

There are two sources of data that we will be using for our analysis - CAIDA [32] and Naval Postgraduate School (NPS); and we will have a source of ground truth to compare against.

The bulk of the analysis was done with datasets from CAIDA, which has active on-going and regularly scheduled data collection since 2008. As such, CAIDA has more historical data than NPS has. In the initial analysis of this data, results obtained from CAIDA were counter-intuitive to expected results. The analysis showed that the amount of change that was detected by the measurement was unusually high. It was in the range of 33% for *vsd* and 44% for *esd*, This reflects the network changes from one cycle of data collection to the next, as discovered by CAIDA's probes when they traversed from their vantage points to their chosen destinations within Egypt.

A program was then quickly put together by NPS researchers to send probes in a different fashion from CAIDA's, and the results created the NPS dataset. These results were found to correlate to the measure's expected results. This helped to verify the intuitions held for the measures and also shed new light on CAIDA's data collection method. Hence forth, the data from this program is for convenience called NPS data.

4.1.1 CAIDA data

These datasets contained data collected by a set of Archipelago (Ark) monitors which are globally distributed [33]. These monitors are located at vantage points and they act as collection stations for the probes that are sent. There are 77 such monitors, and they are located on all continents of the Earth, except for Antarctica [34].

Ark is CAIDA's current active measurement infrastructure. Its development was motivated by the need to reduce the effort required to develop and deploy sophisticated large-scale mea-

surements and to go towards a community-oriented measurement infrastructure, which allows collaborators to run their measurements on security-hardened distributed platforms. Of the several measurement datasets obtained, of interest to our analysis was the Internet Protocol version 4 (IPv4) Routed /24 Topology Dataset [8] explained below.

Ark allows a coordinated team probing effort to obtain large-scale traceroute-based topology measurements. Currently, monitors are grouped into three teams and the measurement work is dynamically divided up among team members. With this parallelized process, a traceroute measurement to all routed /24's [5] is obtained in about two to three days for a team of 17-18 monitors probing nine and a half million /24's. We call this period of two to three days of data collection a cycle.

Data is collected by sending probes continuously from randomly selected vantage points in these monitors to destination IP addresses. From each IPv4 /24 prefix on the Internet, a destination is randomly selected, such that a random address in each prefix is probed approximately in every cycle. A complete cycle of probed data collection would take two to three days to be completed. For each cycle, this uncompressed data averaged $\approx$ 3.5 GB in size and can be parsed by appropriate tools such as the sc_analysis_dump tool [6] included in the scamper [35] distribution package.

4.1.2 NPS Data

This is data collected by a Python program that was put together to obtain data on specific network segments in a deterministic fashion. The program sends out hourly probes from fixed vantage points to selected destinations. The initial motivation to collect this data was to verify certain intuitions that we had on why the *esd* and *vsd* measures did not give readings as expected when applied on CAIDA's data. Although the Ark infrastructure is also utilized, its methodology differed from CAIDA's in that it probes a specified destination network segment from fixed vantage points. Recall that CAIDA probes randomly selected /24 addresses and from random vantage points as well. With the randomness caused by the source and destination points removed, the traceroutes between cycles of probing were expected to be more similar. This was verified when the *esd* and *vsd* measures applied to this dataset gave a low reading

[5] An IPv4 address is a 32-bit integer value. /24 is the prefix of the IPv4 network starting at a given address, having 24 bits allocated for the network prefix.

[6] This utility provides a dump of traceroute data in a textual format that is easily parsed by scripts.Each line in the output contains a summary of a single trace. This includes the interfaces visited and the delay of each response. The details of the output format can be seen in Appendix 6.3.

which averaged ≈0.6% per cycle, as compared to the high reading of ≈40% from CAIDA.

4.1.3 Purdue University's data

The network topology of Purdue University is obtained from network configuration files that were used to configure the network devices within the campus. Router devices were first identified by looking for configuration commands associated with the router configuration. The portions of data that related to router configurations were then extracted. Interfaces of these routers were then consolidated and sorted by subnets. Assuming that interfaces on the same subnet were connected, the inter-router connectivity was derived. Thus, Purdue's internal network topology at the router level was obtained.

4.2 Data Selection and Preparation

We describe the data collected by CAIDA to give the reader a feel for the amount of processing involved. A parse of the entire dataset for one cycle collected by a single team gave over 9,000,000 traces, from which we obtained approximately 900,000 vertices and 1,600,000 edges. Vertices, in our context, would represent interfaces (of routers), and edges would be connections between interfaces that were discovered by the traces. Due to the immense size of the Internet, to see a one percent change in our measurement readings, we would need a change in $\approx$ 18,000 vertices or $\approx$ 32,000 edges to occur. With a dearth of events that could bring about such an impact at that level, only specific areas of the Internet were picked for a closer look, so that the readings would be more significant.

A study [4] was previously done on the disruption of Internet communications in several North African countries. This disruption was in response to civilian protests and threats of civil war which occurred in the first months of 2011. With this as a lead, we chose Egypt and Libya as areas of interest to determine if our measurements could detect these events. Purdue University was also chosen for a comparison as ground truth. Details of these are in Chapter 5.

4.2.1 Preparation

The output from sc_analysis_dump tool[6] is a list of traceroute-like data. An entry in this list closely resembles the output from a traceroute, where measurements of transit delays of packets across an IP network are taken together with the history of the route recorded as RTT of the packets received from each successive router along the route. As we are concerned with how the network is mapped, rather than its performance, we concern ourselves only with the router

interfaces and disregard the RTT measurements. From one such entry, we have a sequence of interfaces in a fixed order.

Representing the interfaces as vertices, we "string" up the interfaces in order. Thus, we have an edge from the first interface in the sequence to the second one, another edge from the second interface to the third one, and so on. So, from each entry in the list of traceroutes, we obtain a path. Doing likewise for the entire list of entries, we obtain several paths, and by forming a union of all these paths (discarding edge and vertex multiplicities), we build a graph. This graph is a snapshot representative of the network map discovered by the probes in that cycle. The *vsd* and *esd* measures allow us to compare two such graphs.

4.2.2 Challenges

Despite improvements from the use of the Paris traceroute as mentioned in Section 2.2.3, there exist routers that do not respond to the probes. Some of these routers drop the probing packets completely, while others choose to pass on the probing packets on to the next router (en-route to the intended destination) but still do not respond to the probes themselves. We refer to the former as probe-dropping routers and the latter as non-responding routers.

The behavior of these routers results in several cases where we obtain incomplete traces. In the first case, where a router drops the probing packets completely, we obtain the traceroute that ends at the router that is upstream of this router. A second case, where we have non-responding immediate routers, results in a trace that is fragmented by one or more intermediate router's non-response. Here, information of the network that is downstream of a non-responding intermediate router is captured since the probe can continue toward its intended destination. An outcome that is a combination of these two cases is also possible. So, we can have a traceroute which is fragmented by non-responding intermediate routers, and which terminates at a probe-dropping router. Of course, there are also cases where the probes traverse non-responding intermediate routers and arrive at their specified destination.

Traces arriving at their destination and not encountering any non-responding intermediate routers en-route are referred to as *complete*; while traces that do not reach their intended destination and/or traverse non-responding immediate routers en-route are referred to as *incomplete*.

In handling such behaviors, only deterministic interface connections recorded by the traceroute record were taken for analysis. Deterministic interface connections would refer to those connections that have interfaces from which a router has responded. A fragmented traceroute might

be seen as comprising more than one separate path, so the resultant graph constructed from the union of such paths could thus be disconnected.

203.181.248.60	203.181.248.60	203.181.248.60
203.181.249.21	203.181.249.21	q
203.181.102.129	203.181.102.129	203.181.102.129
118.155.197.1	118.155.197.1	118.155.197.1
203.181.100.126	203.181.100.126	203.181.100.126
59.128.2.210	59.128.2.210	59.128.2.210
65.19.143.9	65.19.143.9	65.19.143.9
72.52.92.37	72.52.92.37	72.52.92.37
72.52.92.233	72.52.92.233	72.52.92.233
216.66.77.102	216.66.77.102	216.66.77.102
209.152.158.18	209.152.158.18	209.152.158.18
q	209.152.158.18	
q		
q		
209.152.158.18		

Figure 4.1: Comparison of traceroutes with same source and destination addresses[7].

An example of different router behavior is illustrated in Figure 4.1. Here, we have three traceroutes obtained with the same source and destination interfaces but at different times. The outputs are different, and this difference shows the unreliability of traceroute probes. Also, when probing poorly connected regions of the Internet, such as Libya and Egypt in our case, packet loss is also a possibility. It is interesting to note that some routers respond at certain times and not at other times, as observed in the second hop of Figure 4.1. By comparing similar traceroutes, further information could be gleaned. By matching information from these traceroutes, and taking into account similarities in intermediate routers' interfaces as well as their RTT, the interfaces of non-response routers could be inferred with some level of certainty. In our example, the hops taken by the traces are largely similar. The non-response of the second hop in Figure 4.1 could probably be inferred from the other two traceroutes.

4.3 Analysis

In the initial analysis of CAIDA's data, it was observed that the comparison of graphs in time (which represent snapshots of the Internet in general) gave a high reading on the *esd* and *vsd*,

[7]These data are extracted from fields 14 and onwards of recorded traceroutes. The value 'q' denotes no response at that particular hop. Refer to Appendix 6.3 for details.

meaning very large changes in the topology. Investigations revealed that part of this was attributed to the way in which the data was collected.

Recall that CAIDA randomly selects a destination from each routed IPv4 /24 prefix for probing and that probes are then sent to these selected destinations from randomly selected vantage points. The effects of probing a selected /24 destination subnet from randomly selected vantage points are described in Section 5.2 where we use NPS data for comparison with that of CAIDA's. We shall now investigate the effect of probing randomly selected destinations in a /24 destination subnet. Here, two probes sent from the same vantage point to the same /24 destination subnet could give different results despite taking the exact same path to the destination subnet. The reason for this was that the final destination interface would probably be different due to the random selection process—an artifact of CAIDA's probing methodology, since CAIDA's goal is to discover as much of the real topology as possible. As a result, the difference between the graphs is artificially increased, and the measure reflects as such with a higher reading. To mitigate this effect, all of our analysis is conducted with the exclusion of this last destination interface.

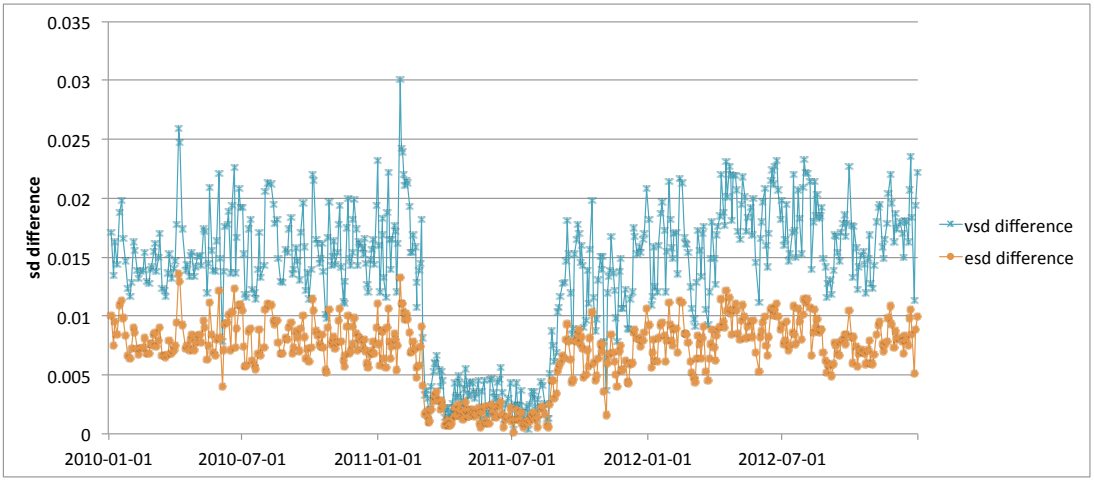


Figure 4.2: Effect of randomly selected last hop on reading differences.

Figure 4.2 is a plot of inter-cycle *vsd differences* of Egypt and Libya ASes for the years 2010 to 2012 using CAIDA datasets. The *vsd difference* plotted in Figure 4.2 is given by the difference between inter-cycle *vsd* readings which includes the randomly selected last destination interface and the difference between inter-cycle *vsd* readings which excludes the randomly selected last destination interface. i.e., $vsd\ difference = vsd_{include\ random\ destination} - vsd_{exclude\ random\ destination}$. We do likewise for the *esd*. It is observed that both the *vsd difference* and *esd difference* are

positive throughout the entire period, indicating that the inclusion of the randomly selected final destination does indeed result in an increase in the measurement readings. The distinct drop in measurements that occurred in February 2011 will be discussed further in Section 5.1.

4.3.1 Spatial

To observe how different segments of a network behave, we can dissect the network with different levels of granularity. The level of granularity could depend on the size of the desired network segment(s). For our case where the segregation is done at a country level, samples of ASes representative of a particular country are first obtained. To do so, the registered organizational names of ASes matching the chosen country are obtained from the Regional Internet Registry (RIR). [8] We thus have the corresponding ASNs for these ASes. Prefixes, as announced by these ASNs, are obtained by observing the global Border Gateway Protocol (BGP) [9] routing table as provided by [36]. Only the /24 prefixes within the larger aggregates announced by the ASNs are considered in the analysis.

To go further in depth into the analysis, the topology external and internal to an identified AS can also be studied. With the ASes of a country identified, analyzing these would be analogous to looking at the topology external and internal to the country.

In technical terms, this is achieved by bisecting the traceroute data. The first part consists of traces from the source to the AS, and the second, the point of entry from the AS to the destination. With this information, the measure is effectively "broken down," such that the contribution made by the two parts, internal or external, can be easily identified.

Henceforth in our research, references to internal and external networks are from the perspective of the ASes, and combined network refer to the convolution of both these networks.

Here are some perspectives that an analysis can offer. Depending on the desired analysis, a combination of perspectives is also possible.

4.3.2 Temporal

To observe how a network behaves in time, snapshots of the network can be taken at time intervals. In fact, an example is previously seen in Figure 4.2 where we discussed the effects

[8] An organization that manages the allocation and registration of Internet number resources within a particular region of the world. Internet number resources include IP addresses and ASNs.

[9] The protocol which is used to make core routing decisions on the Internet; it involves a table of IP networks or "prefixes" which designate network reachability among ASes

of including randomly selected destinations in our analysis. We consider readings in the time intervals t_0 to t_1, t_1 to t_2, and so on until the last time interval t_{n-1} to t_n, where t_n is the desired end time. In so doing, we obtain readings in the desired time range t_0 to t_n with a step size of one unit. In this case, a unit is the time interval from one cycle of CAIDA's probing to the next cycle.

Building on this, a snapshot from one point in time t_i could be compared to another in any other point in time t_j in a non-sequential fashion. In other words, the network at time t_0 is compared to the network at time t_1, t_2 and so on until t_n; the network at time t_1 is compared to the network at time t_2, t_3 and so on until t_n; and doing likewise, all other values of t_i are iteratively compared to t_j. With this, a graph can be constructed to facilitate the study of possible relationships of the measure from one point in time to any other time. Note that since the measure is symmetric, the entire space of size n^2 need not be computed, rather just $n^2/2$. We will briefly illustrate the use of non-sequential time comparison in Chapter 5 of our analysis.

Comparison using windows (i.e., time frames) of varying sizes can also be done. To perform a comparison for a window of a specified size, all cycles within that window are overlapped, giving a bigger and denser snapshot. Recall that each cycle is represented as a graph, so the graph representation of the window is the union of the cycle graphs within that window, again, discarding multiplicities. Comparison between windows is then done by comparing this union-of-graphs with the next union-of-graphs and so on. In the analysis, windows of varying durations - daily, monthly, two-monthly (i.e., every two months), and three-monthly (i.e., every three months, or quarterly) were used. Since the duration of one cycle of CAIDA's probing is approximately two and a half days, these inter-cycle plots are loosely referred to as "daily" plots. A monthly plot would be the accumulation of data over several cycles with a calendar month cut-off. To obtain a graph representative of one month, the "daily" cycles for that month would be accumulated by taking the union of these "daily" cycle graphs. The process described is repeated to obtain the data for each subsequent month, after which the graph of one month is compared to the one of the next month, and so on. Likewise, a two-month and three-month reading is the accumulation of cycles within cut-offs of two and three calendar months respectively.

4.3.3 Frequency and Probability Distribution

The frequencies of different class intervals can be tabulated and presented in the form of a histogram. The class intervals in this case would be ranged values of the measures' indices.

By normalizing these frequencies, we can obtain the approximate probability distribution of the measures' indices. In this, the probability for each class interval is easily observed, with the total area equalling one.

The histogram helps us to quickly identify the shape, mode and spread of the readings quickly. The shape would indicate how the classes are distributed; the mode would be the most frequently-occurring class; and the spread shows how different the values are from each other and from the mode.

We will use this in Chapter 5 of our analysis.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Case Studies and Results

In this chapter, we detail the results obtained by applying the *esd* and *vsd* measures in various ways. In all cases, the measures are applied to graphs that represent inferred segments of identified Internet topology. The chosen segments roughly represent countries or institutions.These specific segments were chosen due to interesting live events that were likely to exhibit significant topological changes and temporal deviations. Taking the lead from [4], the Egyptian and Libyan ASes were chosen as representatives of their respective countries' network-structuring bodies.

5.1 Egypt/Libya Network from CAIDA Data

We focus on a specified area of the Internet—the Egyptian and Libyan network. Some results from the application of *esd* and *vsd* measurements over a three-year duration probing Egypt/Libya are presented and discussed.

Three-years' worth of CAIDA datasets, as detailed in Table 5.1, were analyzed. The identified Egyptian and Libyan ASes that were chosen for this case study are as shown in Table 5.2. Traces with destination addresses that fell within the prefixes of these identified ASes were then extracted and analyzed.

	Traces processed	Distinct edges discovered	Distinct vertices discovered
Containing identified Egypt/Libya Prefixes	595,783	27,521	8,700

Table 5.1: Statistics of CAIDA dataset used for three-year study of Egypt and Libya.

The ASes are identified, and the prefixes they originate in the global BGP table as inferred by Routeviews [36] are shown in Table 5.2.

By focusing on these ASes, an inferred topology of the networks encompassing Egypt and Libya was constructed. In our context, the address prefixes of these ASes are representative of the countries mentioned. Note that other ASes and prefixes reside in Egypt and Libya. Our goal was not to be exhaustive, but rather to focus on several large networks known to be in these countries that would serve as representative destinations to probe.

ASN 13388, EGYPTIAN-TELEPHONE - Egyptian Telephone:	65.214.64.0/21 208.103.192.0/19 216.138.48.0/20 216.138.56.0/21 216.158.112.0/20
ASN 25364, EgyptCyberCenter-AS:	81.29.96.0/20
ASN 21003, GPTC-AS Libya Telecom	41.208.64.0/18 62.240.32.0/19 41.252.0.0/14

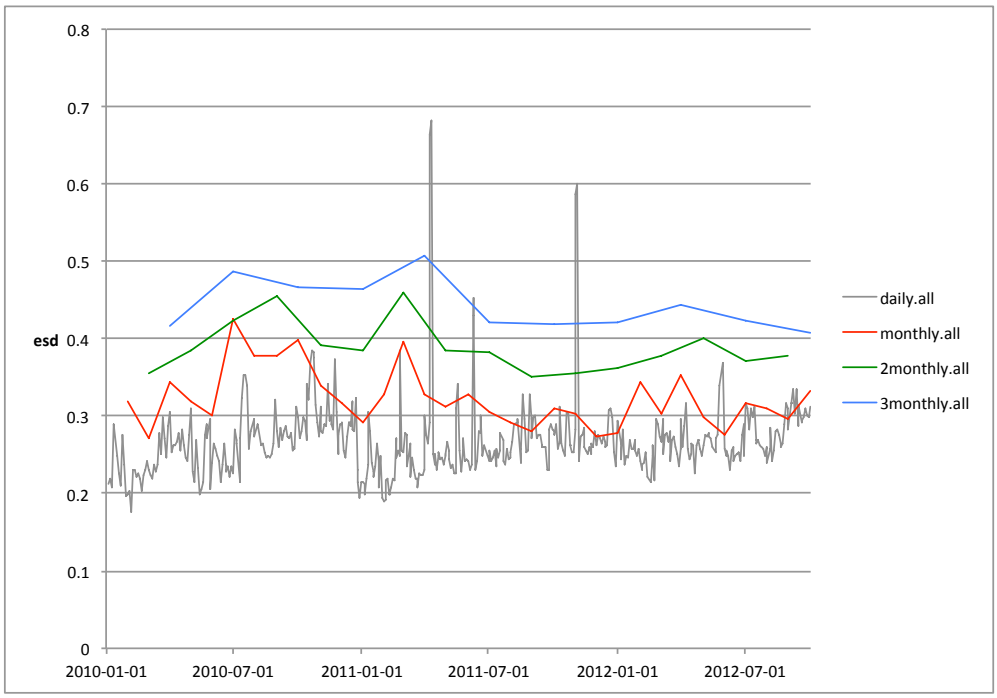
Table 5.2: ASes of Egypt and Libya that were used in our case study.

5.1.1 Temporal

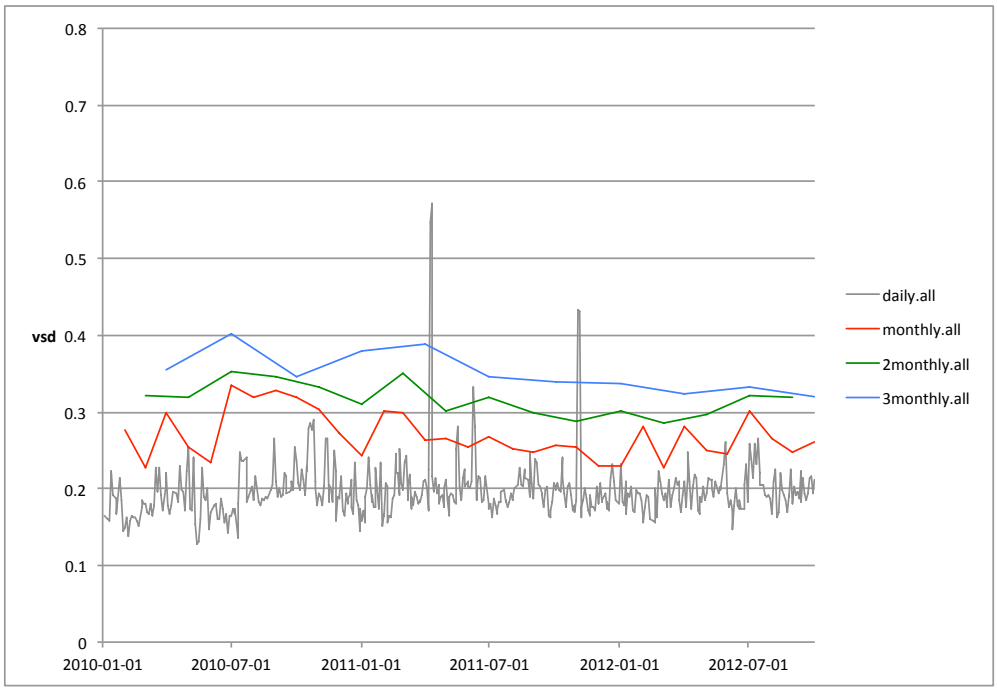
Readings for "daily" plots (i.e., *esd* and *vsd* measures that compared cycle-to-cycle data) were observed to fluctuate substantially. To investigate the effects of using different window size for comparison, monthly, two-month and three-month (i.e., quarterly) plots were constructed as seen in Figure 5.3.

Recall from Section 4.3.2 that a window size is increased by taking the union of the aggregated cycles within that window. It was observed that when the window sizes were increased in duration, the readings also increased. This happened for both *esd* and *vsd* readings. With a window size of a longer duration, more vertices and edges are accumulated within one window for comparison. We note here that when making comparisons using CAIDA data, the union of CAIDA cycles yields a better picture due to CAIDA's random monitor selection when collecting data. The effect of CAIDA's random monitor selection is discussed further in Section 5.2. So, if similar interfaces and connections come and go, their accumulation over time would give a zero net effect on *esd* and *vsd* measures. Likewise, accumulation of dissimilar interfaces and connections would result in an increased net effect on *esd* and *vsd* measures. The results seem to suggest the latter case.

It could also be the case where the window size chosen was yet to be large enough to obtain the zero net effect. Notice that as the window size used for comparison increases, the readings for both *esd* and *vsd* increase (as seen in Figure 5.3), but the magnitude of fluctuations decreases (as seen in Figure 5.2), which is intuitive. Here, $fluctuation_{esd} = max(esd) - min(esd)$ and $fluctuation_{vsd} = max(vsd) - min(vsd)$. We varied the window size used for comparison, and plotted the magnitude of the fluctuations in readings per average cycles per given window size, against the different window sizes. The average cycles per given window are as shown in



(a) *esd*.



(b) *vsd*.

Figure 5.1: Readings for network (i.e., internal and external combined) of Egyptian and Libyan ASes.

Table 5.3. Thus, the magnitude in fluctuations is normalized, and this was done in order to capture the average fluctuation per cycle across different window sizes. Due to its exponential nature, the results are illustrated in Figure 5.2 using a logarithmic graph for the vertical axis. The results show that the decrease in fluctuation was most significant when the window size was increased from "daily" to monthly, where the readings drop by almost two orders of magnitude.

Window size (duration)	Average cycle(s) per given window size
"daily"	1.0
monthly	13.9
2-monthly	27.8
3-monthly	41.8
6-monthly	83.5
9-monthly	125.3
yearly	167.0

Table 5.3: Average cycle(s) per given window size.

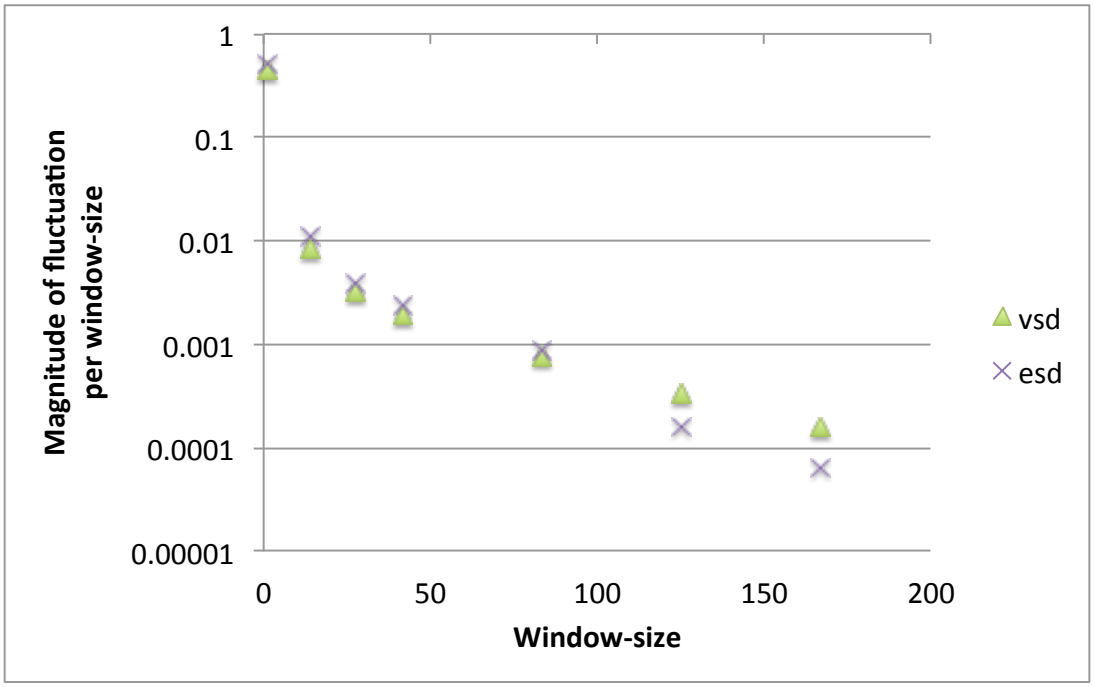


Figure 5.2: Effect of comparison window size(duration) on fluctuations in readings.

The results from Figure 5.2 seem to support the hypothesis that a near-zero net effect could be achieved by choosing a large enough window size.

5.1.2 Spatial

Here we separate the networks into those internal and external to the ASes.

In Figure 5.1a, we observed fluctuations in the plot for the combined network (i.e., internal and external) without distinct anomalies. However, in Figure 5.3 where we look at the plots of internal and external networks separately, pronounced changes were observed for the internal network, as compared to that for the external network. That is because even big changes in a few ASes, such as those in Egypt and Libya, will not have a big effect on the bigger picture of the Internet.

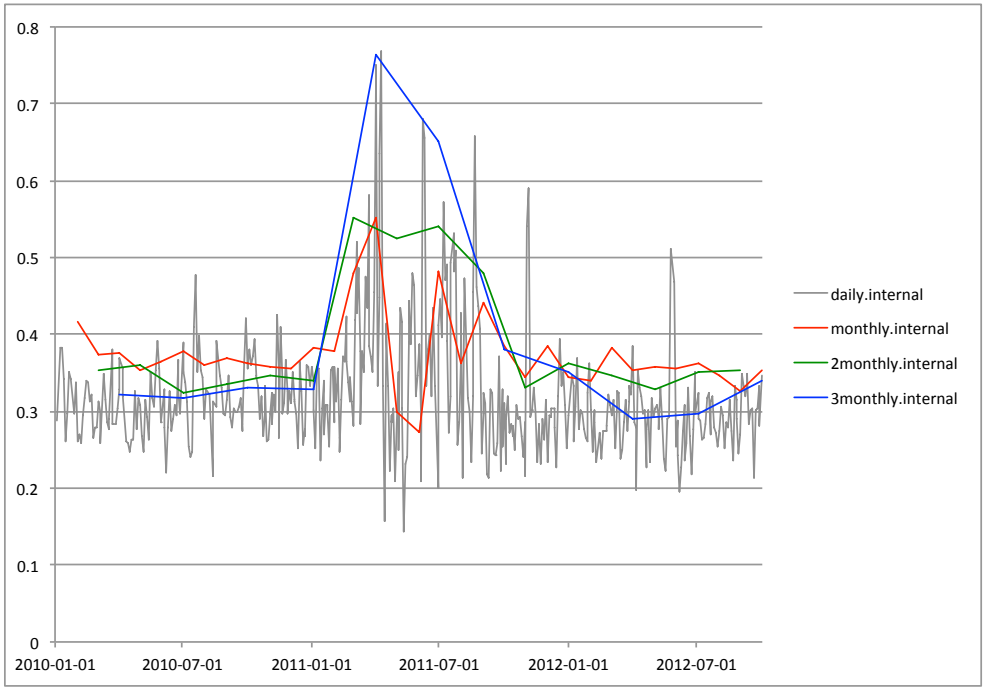
In fact, the plot representing the external network closely resembles the one from the combined network. This observation was verified in that there were significantly more external edges as compared to internal ones, and is shown in Figure 5.4. One can think of the external network changes as noise that masks the changes happening internally.

An interesting observation was made in Figure 5.3a. When there were changes in the network, such as during the Arab Spring, the *esd* readings were higher when a larger window size was used for comparison; and when the network was relatively stable, the converse took place instead. i.e., *esd* readings were lower when a larger window size was used. This shows that the change was not merely due to the drop in the number of vertices, but rather, in the instability of the network where different vertices came in and out of the network. This instability is exactly what we were hoping that our measure would depict.

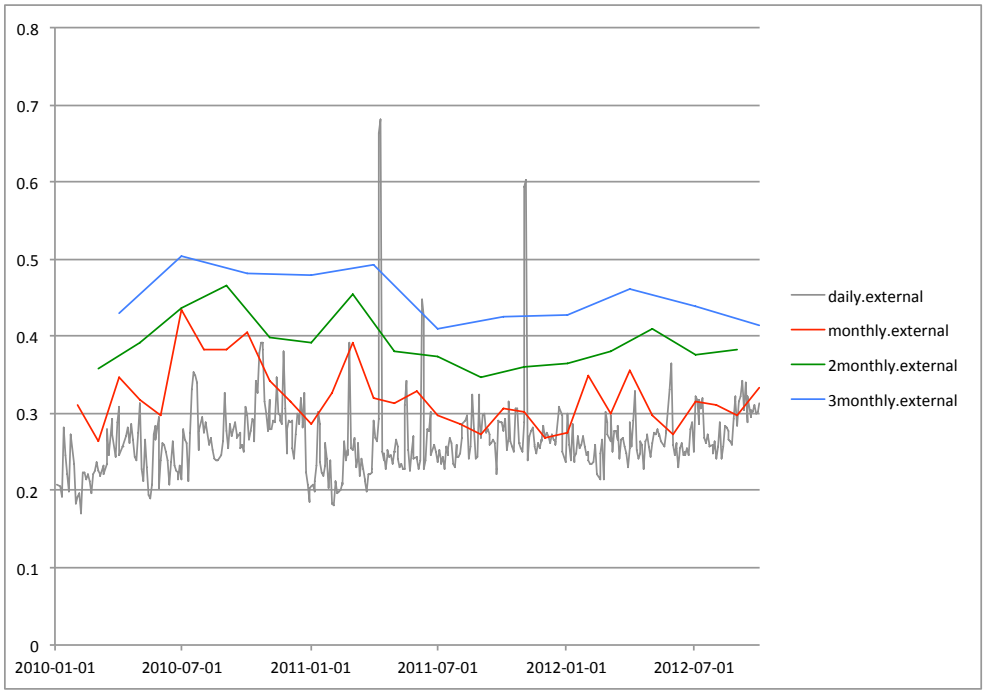
Delving deeper, we looked at the absolute rate of change of the index. This was obtained by plotting the absolute difference between two sequential readings. In other words, it gives us the magnitude of the time derivative of the index. Illustrated in Figure 5.5 is this rate of change with different granularities—daily, monthly, two-month and three-month basis.

The rate-of-change plot helps to identify times of interest by accentuating the period of time in which changes in *esd* occur.

Next, we take a brief look at network comparisons in non-sequential time. Thus far, we have been looking at comparisons in sequential time, meaning that we compared the snapshot of a network to another snapshot in a chronological manner. We now explore network comparisons between one cycle (in time) to every other cycle (in time), where each cycle represents a time period of about three days. This comparison is done in order to consider the case of



(a) Internal.



(b) External.

Figure 5.3: Readings of *esd* for different network segments (i.e., internal and external) of Egypt and Libya.

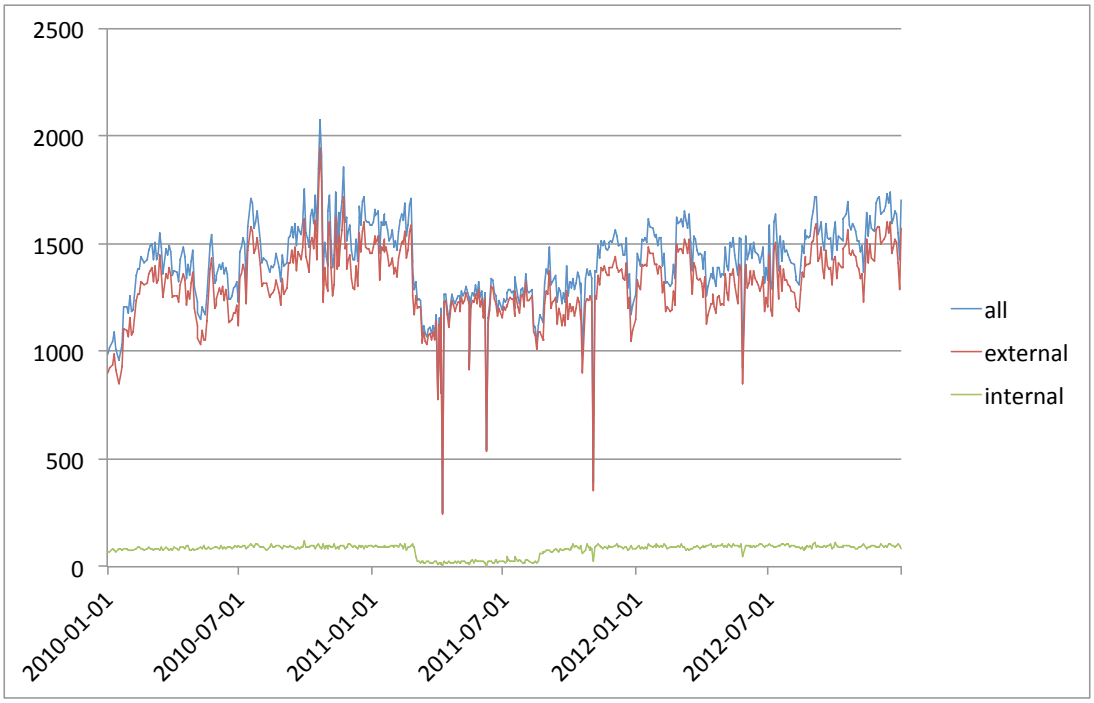


Figure 5.4: Edge counts for Egypt and Libya AS from 2010 to 2012.

constant changes in the network or merely consecutive changes in time. Considering the data in Figure 5.3, we notice the oscillating effect in the graph of the daily reading, so it is interesting to see if the network just bounces between two states or if it really is that living organism that keeps changing. We do so for the prior time intervals mentioned, and obtain Figure 5.6. The horizontal axes have been indexed such that they represent the chosen unit (i.e., "daily", monthly, two-month or three-month) in a chronological fashion. Taking the two-month plot for example, the first two-month period (i.e., January 2010 to February 2010) is indexed as 0, the second two-month period (i.e., March 2010 to April 2010) is indexed as 1, and so on.

We observe that there is a distinct "ridge" that corresponds to the period of February to March 2011. The "ridge" represents a collection of points that have higher than normal *esd* readings as compared to all other readings in the three-year study. This depicts the period of time in which significant changes in the network were occurring within our area of interest, which we correlate with the Egyptian revolution. Also, notice that indeed there are changes of about 40% across the whole interval, when comparing any day to any other day. This confirms that there are constant changes in the Internet, and it is not the case that the Internet bounces back and forth between certain states.

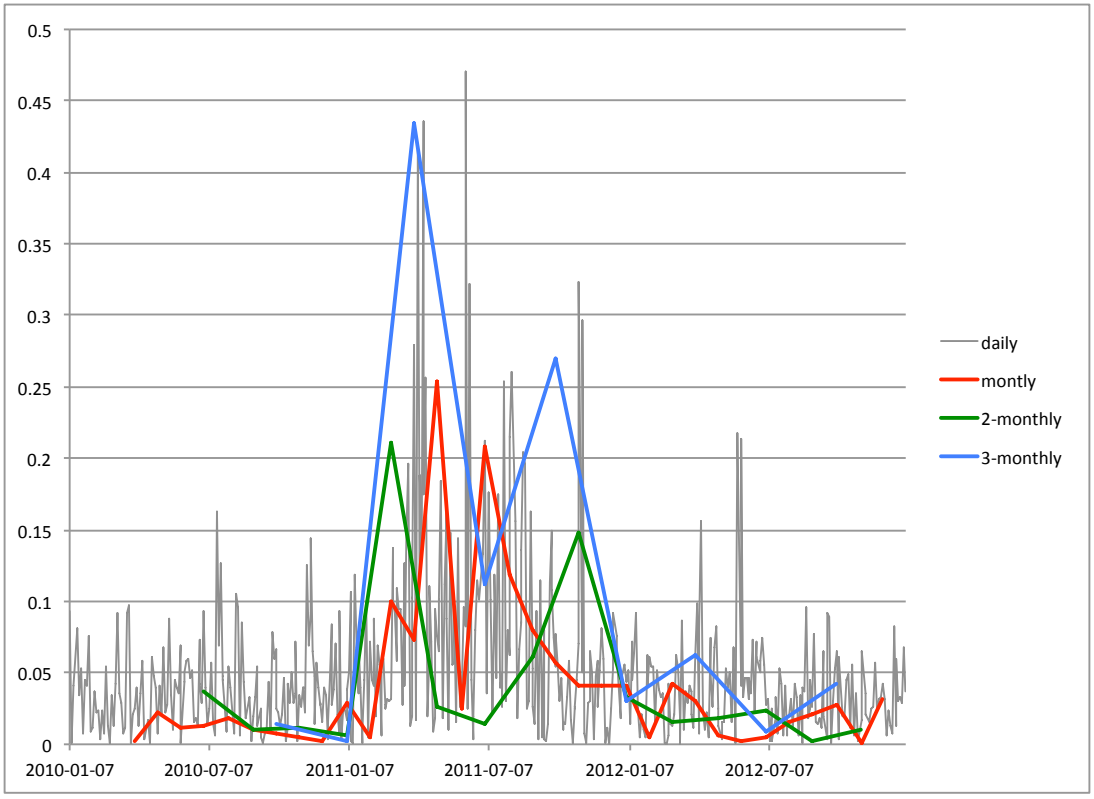


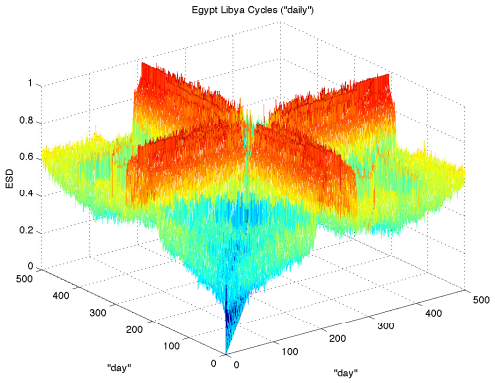
Figure 5.5: Rate of change of *esd* for internal network of Egyptian and Libyan ASes.

5.1.3 Frequency Distribution

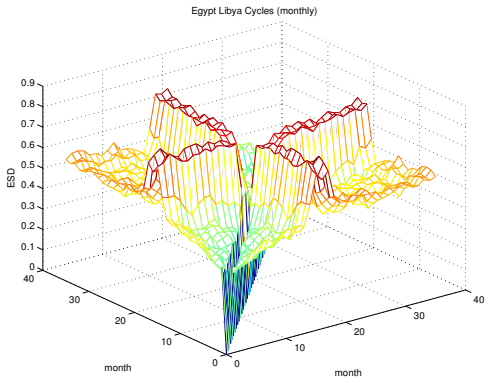
In Figure 5.7, we observe how the *esd* readings are distributed. A distinct mode is observed at about 0.25 throughout the three years. The shape of the histogram for the years of 2010 and 2012 bear close resemblance to each other, while that of 2011 is remarkably different. This irregularity arouses the urge for further investigation.

The readings are separated into internal and external networks in order to investigate which one of the two (i.e., internal or external) networks has greater influence on the combined network readings. The "breakdown" histogram in Figure 5.8 shows that the histogram of the external network resembles the combined one in Figure 5.7.

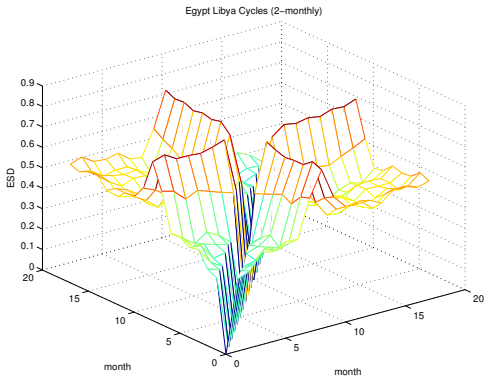
As seen earlier in Figure 5.4, due to the significant larger proportion of edges that were contributed by external networks, the *esd* readings for external networks were found to closely resemble that of the combined network. At the same time, the readings for the internal network were more susceptible to changes inside the Egyptian and Libyan ASes of course, as these took place in a comparatively smaller network.



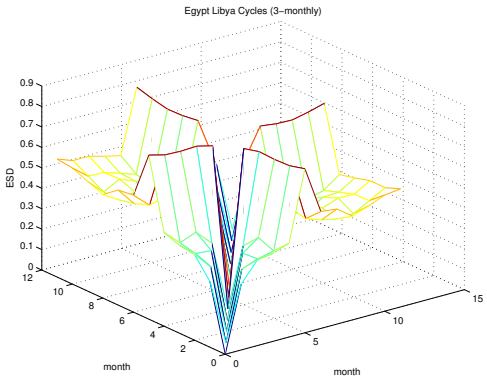
(a) “Daily”.



(b) Monthly.



(c) 2-Month.



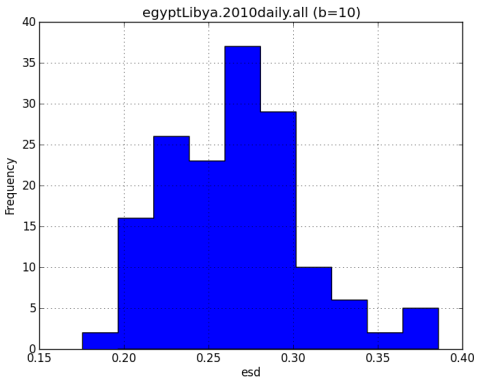
(d) 3-Month.

Figure 5.6: Non-sequential *esd* readings for internal network of Egypt and Libya.

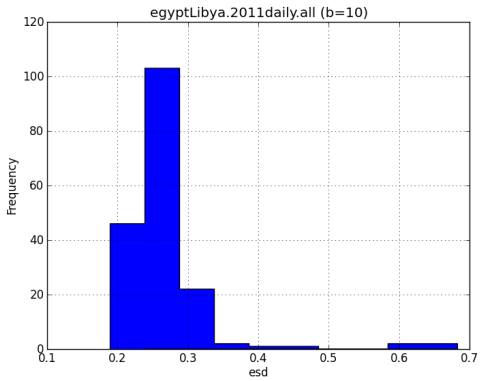
By decomposing Figure 5.7 spatially into internal and external networks as illustrated in Figure 5.8, we get a clearer view of the fluctuations in the readings for the internal network. Once again, the “noise” from the external network masks out changes in the internal network. The changes in the latter provide a good indication to social and political events that were happening to Egypt and Libya, and the figures above affirm that the events occur around the first months of 2011, which coincides with the Arab Spring timeline.

5.2 Egypt Network from NPS Data

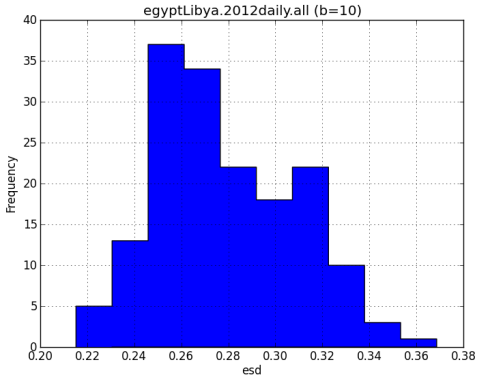
In this section, we investigate the effects that randomly selected vantage points have on our measurements, which motivate the way NPS collects data. The prefixes used in this context are as shown in Table 5.4.



(a) 2010.



(b) 2011.



(c) 2012.

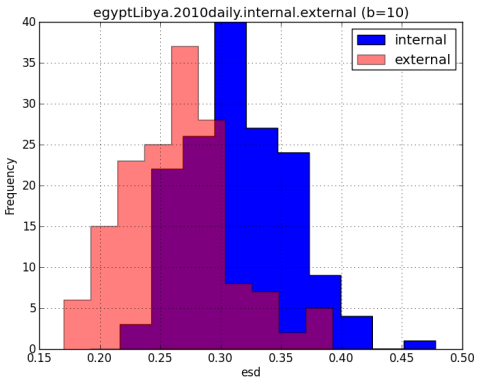
Figure 5.7: Frequency distribution of "daily" *esd* readings for three years (combined network).

ASN 13388, EGYPTIAN-TELEPHONE - Egyptian Telephone:	65.214.64.0/21 208.103.192.0/19 216.138.48.0/20 216.138.56.0/21 216.158.112.0/20

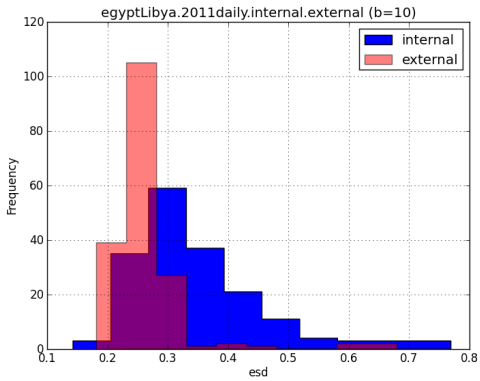
Table 5.4: AS of Egypt that was used by NPS.

The data was collected both by CAIDA and NPS over the same period of four weeks for the same fixed AS. Statistics of the data are as shown in Table 5.5

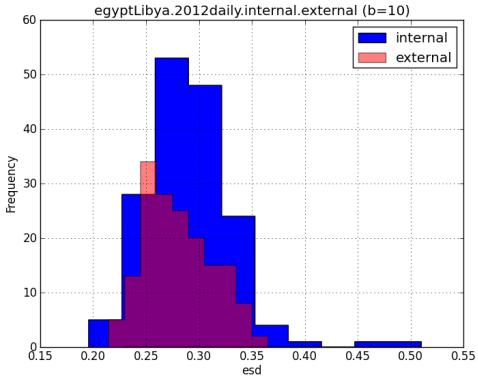
Using the Ark infrastructure, probes were sent from pre-determined vantage points to specified destinations in the chosen AS above. This method of data collection differed from that of CAIDA's in that the vantage points and the destination address were not randomly chosen.



(a) 2010.



(b) 2011.



(c) 2012.

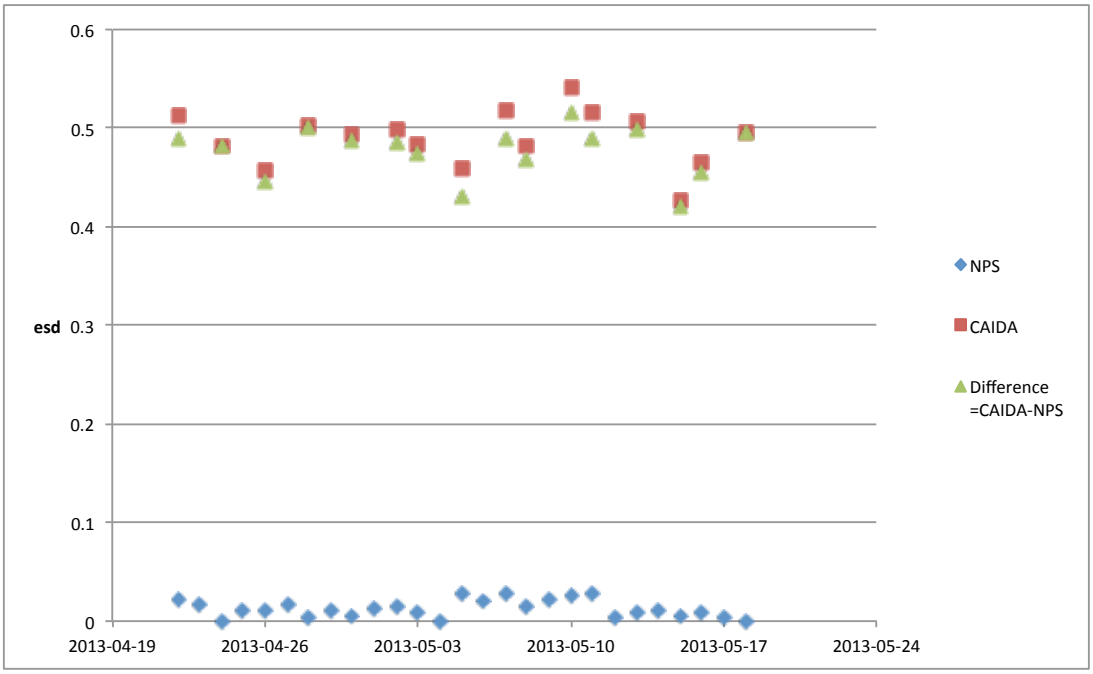
Figure 5.8: Frequency distribution of "daily" *esd* readings for three years (internal and external network breakdown).

	Number of traces processed	Distinct edges discovered	Distinct vertices discovered
NPS	18,173	571	410
CAIDA	1,455	2,212	1,295

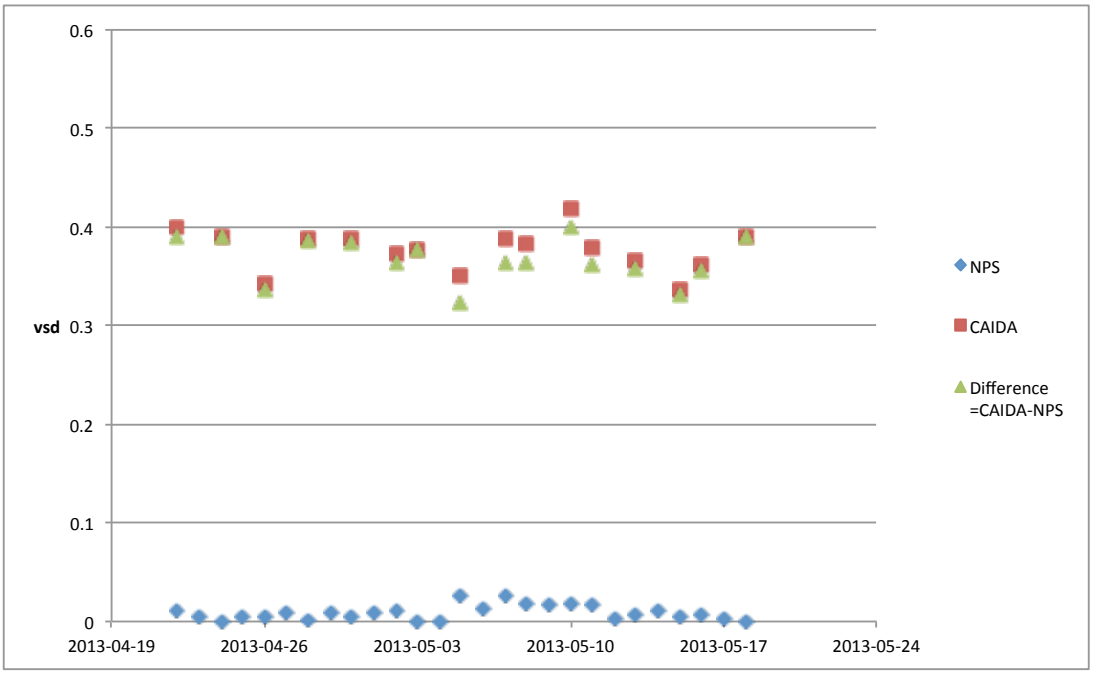
Table 5.5: Statistics of dataset used by NPS for four-week study of Egypt.

Recall from Section 4.1.2 that probes were sent out hourly for NPS's collection method. To have a better comparison with similar data from CAIDA's "daily" window size, NPS's hourly data was first aggregated to a daily window size. Comparing NPS data versus CAIDA, we obtain the readings of *esd* and *vsd* of Figure 5.9.

From Figure 5.9, we observe that the *esd* and *vsd* measurements from CAIDA are constantly



(a) *esd*.



(b) *vsd*.

Figure 5.9: Effects of randomly chosen vantage points on measurements.

higher than that obtained from NPS data. For sure, some of the increase in measurement read-

ings is attributed to the random selection of vantage points. The magnitude of this increase is depicted by the "Difference" plots, which takes the magnitude of the difference between CAIDA plots and NPS plots. For *esd* and *vsd*, these differences average approximately 48% and 37% respectively, which is very substantial.

We have seen in Section 5.1.1 that the fluctuations in readings can be decreased by the use of larger window sizes used for comparison. It remains to be seen if we can reduce this artificial increase in measurements, which is attributed to using randomly selected vantage points, by using larger window sizes. However, we do not have sufficient NPS data over a longer time period to verify this.

5.3 Purdue University - CAIDA versus Ground Truth

For this section, we compare the topology derived from CAIDA's datasets with the topology that is derived from Purdue University's network configuration files. Henceforth, we refer to the latter as ground truth. The comparison is done at a different granularity level, but in the same way as before, since Purdue's data gives us the router-level topology instead of the usual interface level ones with which we have been dealing.

By sieving out the router configurations as described in Section 4.1.3, an interface-to-router hostname lookup table was constructed. This lookup table would allow us to obtain a unique identifier of the router (such as its hostname) that was associated with a given interface.

From CAIDA's data, we extracted only the interfaces that were internal to Purdue's prefixes as listed in Table 5.6. Then, using the aforementioned lookup table, these interfaces were translated to their corresponding router hostnames to obtain a router-level topology.

Purdue University	128.210.0.0/16 128.211.0.0/16 128.10.0.0/16

Table 5.6: Prefixes used by Purdue University.

Corresponding graphs were constructed from these two sets of data (i.e., topology obtained from Purdue's configuration files and Purdue's internal network derived from CAIDA's datasets), and the *esd* and *vsd* measurements were applied. Here, again, the traces from CAIDA were trimmed and only the internal data to Purdue were used.

As Purdue's configuration file was collected on January 6, 2011, the corresponding month's dataset from CAIDA was taken for comparison.

From Purdue's configuration files, 178 vertices and 2975 edges were obtained. While from CAIDA, 160 vertices and 78 edges were obtained. Note that vertices represent routers and edges represent pair-wise connections between routers. Applying our measurements between these two sets of data, we obtained *vsd* and *esd* readings of 0.68 and 0.98 respectively. From the measurements, we infer that the internal network of Purdue University, as discovered by CAIDA probes, is quite different from that which is implemented on campus, particularly at the edge level. However, it was expected that we would not discover most of the physical connections. We elaborate on our findings in the following paragraphs.

Denoting the graph representing ground truth as G and the derived topology graph from CAIDA as H, the comparison of these two graphs is visualized and is shown in Figure 5.10. The visualization has been color-coded as follows:

- Red: These are vertices common in both G and H, and constitute 26.47% of all vertices.
- Green: These are vertices present only in H and not in G, and constitute 12.75% of all vertices.
- Blue: These are vertices present only in G and not in H, and constitute 60.78% of all vertices.

The green vertices observed in Figure 5.10 are those with IP addresses, which have not been resolved from our lookup table. These probably represent addresses assigned to Purdue but used as the remote address for equipment not managed by Purdue, e.g., a point-to-poink link with a service provider. As such, they are not captured by the configuration file obtained. The blue vertices represent the routers from the configuration file that are not discovered by CAIDA. A fair amount of such vertices are expected, since the configuration file provides a more comprehensive source of information on Purdue's internal network topology.

Next, we investigate the "ideal" number of aggregated cycles from CAIDA that is required to obtain a topology that is most similar to that which is depicted by Purdue's configuration file. By closest, we mean the comparison of two graphs which give the lowest *vsd* and/or *esd*. From operator feedback, we know that there has not been any significant change in the network configuration for Purdue. We begin by comparing the derived router-level ground truth topology with increasing window size used for comparison. In other words, we compare the ground truth

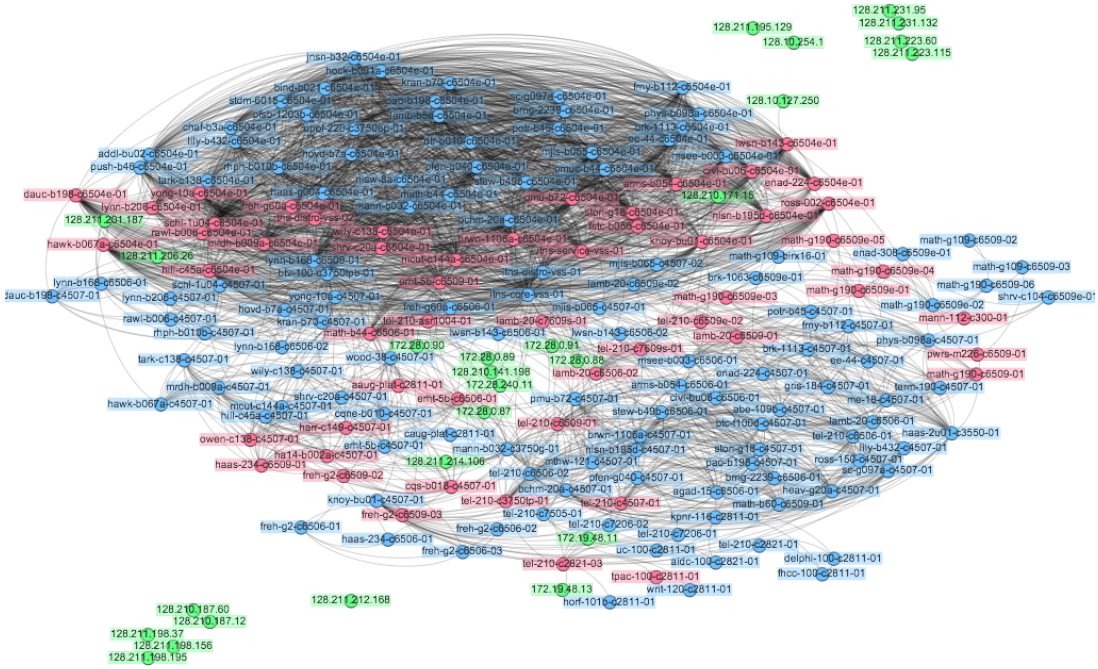


Figure 5.10: Visualization of comparison of two graphs - topology obtained from Purdue's configuration files and that obtained from CAIDA.

with one cycle, then ground truth with the union of two cycles aggregated, ground truth with the union of three cycles, and so on. Starting from the first cycle in January 2011 and with a window size of one cycle, we do this comparison iteratively, with window size increasing by one cycle in each iteration. This comparison is done for the entire year of 2011, and the plots obtained for the *vsd* and *esd* are as shown in Figure 5.11.

We observe that *vsd* had the lowest reading when eight cycles were aggregated, after which the readings went up. We postulate that the optimal number of routers within Purdue's network were discovered after eight cycles, after which increasingly many routers outside of the univer-

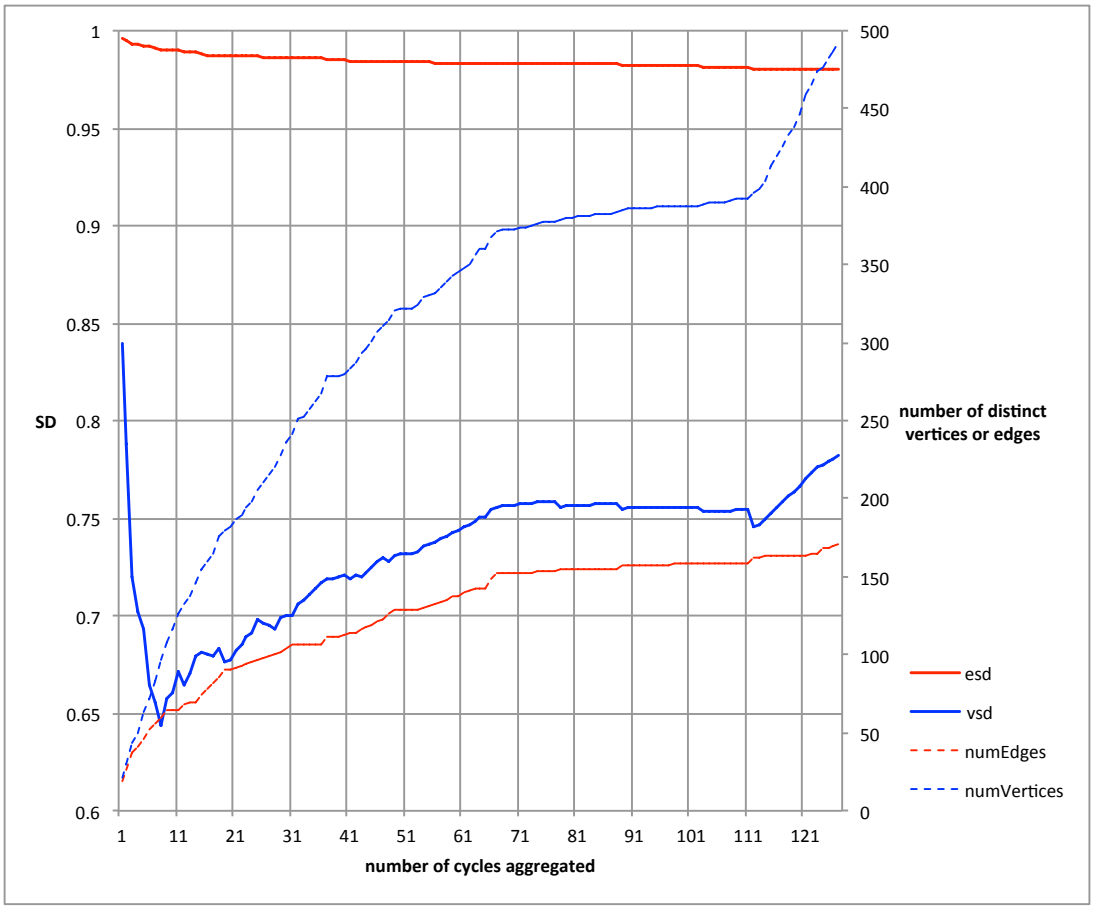


Figure 5.11: Comparison of aggregated cycles from CAIDA with topology obtained from Purdue's configuration files.

sity's jurisdiction were discovered. Meanwhile *esd* was observed to be decreasing slowly as more cycles were aggregated. The number of distinct connections between discovered routers (i.e., edges) was also found to increase at a much slower rate than the number of routers (i.e., vertices) themselves. This is counter-intuitive since routers usually have connections on more than one interface, and we would expect the number of edges to increase at a faster rate than the number of vertices.

CHAPTER 6:
Future Work and Conclusion

In this chapter, we present our findings and provide insights into areas that might require further research.

6.1 Summary

Our goal was to introduce a new and intuitive measurement for comparing graphs that is computationally fast and scaleable to large graphs.

The *esd* and *vsd* measurements were proposed, and they capture changes in edges and vertices respectively. Being normalized, the measurement intuitively reveals whether differences between two graphs were low or high: with zero being lowest (i.e., the two graphs were exactly the same) and one being highest (i.e., the two graphs were completely different). These measures give better insight into why the networks are different than merely looking at counts of vertices and edges. That is, it captures if the networks bounce back and forth between certain stages or constantly evolve, and we found that the latter was true when analyzing specific snapshots of the Internet router interface graph.

By transforming the network-in-question into a graph, and then performing set operations for its set of vertices and set of edges, the measurements can scale up to accommodate the comparison of very large networks. Moreover, involving only simple set operations, the measurement is computationally fast.

We tested these measures on data captured from traceroutes to a combination of a few fixed ASes, and discovered, as intuitively it makes sense, that the *esd* for changes within ASes did not correlate with *esd* outside of these ASes (i.e., the rest of the Internet). We then applied these measures just to the data internal to the union of these ASes that constitute Egypt and Libya, and we noted the correlation between political events happening in that area, and the change in the internal network according to our measures.

6.2 Future Work

We have barely scraped the surface for using these measurements, and there remains much to be explored.

The following are some areas for possible future research.

(1) Perform similar measurements on different datasets.

The bulk of the dataset used in our research uses CAIDA datasets. It would be insightful to consider using other datasets. In so doing, the difference in the datasets could point out where one excels over the other. Their individual strengths might then complement each other to build a more accurate topology for the Internet.

It is also noted that this research utilizes the CAIDA dataset from only one team (i.e., team1). Due to time constraints, we were not able to study the data obtained by other teams. Comparing all team data, or even aggregating all data might serve to provide a richer data source.

(2) Possible correlation between the window size used for comparison and the readings that are taken from the measurements.

In Section 5.1.1, specifically Figure 5.2, it was observed that the fluctuations of the readings were smaller when the window size used for comparison were larger. Ideally, it might be best to aggregate as large a window size as possible to build a "good" picture of the Internet. However, due to the Internet's dynamism and also the nature of the study involved, it might not be practical to do so. Thus, the goal is to find the sweet-spot and strike the balance between having a good enough picture with the smallest possible window size.

In Section 5.1.2, we saw that when there were changes in the network, the *esd* readings were higher when a larger window size was used for comparison; and when the network was relatively stable (i.e., due to the lack of change or growth), the *esd* readings were lower when a larger window size was used. Further research could probably be conducted to study this relationship, which might speed up the process in identifying interest areas.

(3) Non-sequential measurements between any two points in time.

In our research, we have looked mostly at comparing readings taken sequentially in time, briefly touching on non-sequential time analysis as seen in Figure 5.6. Further research could possibly look into "globally" expected network changes, which might shed light to better detection of network changes which deviate from the "global" norm.

(4) Combined measure using both *esd* and *vsd*.

Our research has essentially investigated the effects on *esd* and *vsd* separately. We saw correlations between the two, *esd* and *vsd* behaving similarly for networks, but it is not always the case that one is larger than the other. Because of the big difference between the *esd* and *vsd* for ground truth, we believe that a measure combining the two could be more appropriate. Thus, despite operating on different entities, these two measures could possibly be combined to give a more wholesome result when comparing networks.

(5) Comparison of measures to primitives from standard graph theory.

The primitives mentioned in Section 2.3.1 can be used for comparison with our measures. Such a study could reveal possible correlations between conventional graph theory metrics and that of our measures.

6.3 Conclusion

With the accelerating growth of technology in modern times, connectivity between systems (people included) has exploded at an even faster pace. Again, our research points to the fact that the Internet really evolves, that new vertices and edges appear all the time, and maybe what was known at one point in time is not very relevant at some time later. Much remains to be analyzed from this "connectedness."

Having seen the application of our measurements on the Internet "organism," these measurements could likewise be applied on biological and social networks, due to their similar dynamic nature and large size. Our hope is that the proposed measures serve to advance the broader community's ability to compare large graphs of various types.

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix: SC Analysis Dump Format

```
# ================================================================
# This file contains an ASCII representation of the IPv4 paths stored in
# the binary skitter arts++ and scamper warts file formats.
#
# This ASCII file format is in the sk_analysis_dump text output
# format: imdc.datcat.org/format/1-003W-7
#
# ================================================================
# There is one trace per line, with the following tab-separated fields:
#
#
# 1. Key -- Indicates the type of line and determines the meaning of the
#           remaining fields.  This will always be 'T' for an IP trace.
#
# -------------------- Header Fields ------------------
#
# 2. Source -- Source IP of skitter/scamper monitor performing the trace.
#
# 3. Destination -- Destination IP being traced.
#
# 4. ListId -- ID of the destination list containing this destination
#              address.
#
#      This value will be zero if no list ID was provided.  (uint32_t)
#
# 5. CycleId -- ID of current probing cycle (a cycle is a single run
#               through a given list).  For skitter traces, cycle IDs
#               will be equal to or slightly earlier than the timestamp
#               of the first trace in each cycle. There is no standard
#               interpretation for scamper cycle IDs.
#
#      This value will be zero if no cycle ID was provided.  (uint32_t)
#
# 6. Timestamp -- Timestamp when trace began to this destination.
#
# -------------------- Reply Fields ------------------
#
# 7. DestReplied -- Whether a response from the destination was received.
```

```
#
#      R - Replied, reply was received
#      N - Not-replied, no reply was received;
#          Since skitter sends a packet with a TTL of 255 when it halts
#          probing, it is still possible for the final destination to
#          send a reply and for the HaltReasonData (see below) to not
#          equal no_halt.  Note: scamper does not perform this last-ditch
#          probing at TTL 255.
#
# 8. DestRTT -- RTT (ms) of first response packet from destination.
#      0 if DestReplied is N.
#
# 9. RequestTTL -- TTL set in request packet which elicited a response
#      (echo reply) from the destination.
#      0 if DestReplied is N.
#
# 10. ReplyTTL -- TTL found in reply packet from destination;
#      0 if DestReplied is N.
#
# -------------------- Halt Fields ------------------
#
# 11. HaltReason -- The reason, if any, why incremental probing stopped.
#
# 12. HaltReasonData -- Extra data about why probing halted.
#
#        HaltReason            HaltReasonData
#        -----------------------------------
#        S (success/no_halt)    0
#        U (icmp_unreachable)   icmp_code
#        L (loop_detected)      loop_length
#        G (gap_detected)       gap_limit
#
# -------------------- Path Fields ------------------
#
# 13. PathComplete -- Whether all hops to destination were found.
#
#        C - Complete, all hops found
#        I - Incomplete, at least one hop is missing (i.e., did not
#            respond)
#
# 14. PerHopData -- Response data for the first hop.
```

```
#
#       If multiple IP addresses respond at the same hop, response data
#       for each IP address are separated by semicolons:
#
#       IP,RTT,numTries                     (for only one responding IP)
#       IP,RTT,numTries;IP,RTT,numTries;... (for multiple responding IPs)
#
#         where
#
#       IP -- IP address which sent a TTL expired packet
#       RTT -- RTT of the TTL expired packet
#       num_tries -- num tries before response received from TTL.
#
#       This field will have the value 'q' if there was no response at
#       this hop.
#
# 15. PerHopData -- Response data for the second hop in the same format
#       as field 14.
#
# ...
#
# N. PerHopData -- Response data for the destination
#       (if destination replied).
#
```

THIS PAGE INTENTIONALLY LEFT BLANK

REFERENCES

[1] David Clark. The design philosophy of the darpa internet protocols. In *ACM SIGCOMM Computer Communication Review*, volume 18, pp. 106–114. ACM, 1988.

[2] Erik Nygren, Ramesh K Sitaraman, and Jennifer Sun. The akamai network: a platform for high-performance internet applications. *ACM SIGOPS Operating Systems Review*, 44(3):2–19, 2010.

[3] Cooperative association for internet data analysis. http://www.akamai.com/html/perspectives/index.html.

[4] Alberto Dainotti, Claudio Squarcella, Emile Aben, Kimberly C Claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of country-wide internet outages caused by censorship. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pp. 1–18. ACM, 2011.

[5] Freedom on the net 2012. http://www.freedomhouse.org/sites/default/files/resources/FOTN%202012%20Summary%20of%20Findings.pdf.

[6] Sally Floyd and Vern Paxson. Difficulties in simulating the internet. *IEEE/ACM Transactions on Networking (TON)*, 9(4):392–403, 2001.

[7] Vern Paxson, Jamshid Mahdavi, Andrew Adams, and Matt Mathis. An architecture for large scale internet measurement. *Communications Magazine, IEEE*, 36(8):48–54, 1998.

[8] The ipv4 routed /24 topology dataset. http://www.caida.org/data/active/ipv4_routed_24_topology_dataset.xml.

[9] Lixin Gao. On inferring autonomous system relationships in the internet. In *Global Telecommunications Conference, 2000. GLOBECOM'00. IEEE*, volume 1, pp. 387–396. IEEE, 2000.

[10] Mehmet H Gunes and Kamil Sarac. Inferring subnets in router-level topology collection studies. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pp. 203–208. ACM, 2007.

[11] Adam Bender, Rob Sherwood, and Neil Spring. Fixing ally's growing pains with velocity modeling. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*, pp. 337–342. ACM, 2008.

[12] Ken Keys. Internet-scale ip alias resolution techniques. *ACM SIGCOMM Computer Communication Review*, 40(1):50–55, 2010.

[13] Mehmet H Gunes and Kamil Sarac. Analytical ip alias resolution. In *Communications, 2006. ICC'06. IEEE International Conference on*, volume 1, pp. 459–464. IEEE, 2006.

[14] Mehmet H Gunes and Kamil Sarac. Resolving ip aliases in building traceroute-based internet maps. *IEEE/ACM Transactions on Networking (ToN)*, 17(6):1738–1751, 2009.

[15] Robert Beverly, Arthur Berger, and Geoffrey G. Xie. Primitives for active internet topology mapping: Toward high-frequency characterization. In *Proceedings of the Tenth ACM SIGCOMM/USENIX Internet Measurement Conference (IMC)*, November 2010.

[16] Benoit Donnet, Philippe Raoult, Timur Friedman, and Mark Crovella. Efficient algorithms for large-scale topology discovery. In *ACM SIGMETRICS Performance Evaluation Review*, volume 33, pp. 327–338. ACM, 2005.

[17] Cooperative association for internet data analysis data. http://www.caida.org/data.

[18] Van Jacobson. traceroute. 1989. ftp://ftp.ee.lbl.gov/traceroute.tar.gz.

[19] Paris traceroute, 2013. http://www.paris-traceroute.net/.

[20] Brice Augustin, Xavier Cuvellier, Benjamin Orgogozo, Fabien Viger, Timur Friedman, Matthieu Latapy, Clémence Magnien, and Renata Teixeira. Avoiding traceroute anomalies with paris traceroute. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pp. 153–158. ACM, 2006.

[21] D.B. West. *Introduction to graph theory*. Prentice Hall, Englewood Cliffs, NJ, USA, 2001. ISBN 9780130144003. LCCN 95024773. http://books.google.com/books?id=TuvuAAAAMAAJ.

[22] KS Sergei Maslov and Alexei Zaliznyak. Pattern detection in complex networks: Correlation profile of the internet eprint. *arXiv: cond-mat*, 205379, 2002.

[23] Romualdo Pastor-Satorras, Alexei Vázquez, and Alessandro Vespignani. Dynamical and correlation properties of the internet. *Physical review letters*, 87(25):258701, 2001.

[24] Alexei Vázquez, Romualdo Pastor-Satorras, and Alessandro Vespignani. Large-scale topological and dynamical properties of the internet. *Physical Review E*, 65(6):066130, 2002.

[25] Alberto Sanfeliu and King-Sun Fu. A distance measure between attributed relational graphs for pattern recognition. *Systems, Man and Cybernetics, IEEE Transactions on*, (3):353–362, 1983.

[26] Eric W. Weisstein. NP-Hard problem. from MathWorld–a Wolfram web resource, 2013. http://mathworld.wolfram.com/NP-HardProblem.html.

[27] Christos Gkantsidis, Milena Mihail, and Ellen Zegura. Spectral analysis of internet topologies. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 1, pp. 364–374. IEEE, 2003.

[28] Gary Chartrand, Heather Gavlas, Héctor Hevia, and Mark A Johnson. Rotation and jump distances between graphs. *Discussiones Mathematicae Graph Theory*, 17(2):285–300, 1997.

[29] *Wikipedia*. Complement (set theory) — *Wikipedia*, the free encyclopedia, 2013. http://en.wikipedia.org/w/index.php?title=Complement_(set_theory)&oldid=551274795.

[30] P Erdős and A R&WI. On random graphs i. *Publ. Math. Debrecen*, 6:290–297, 1959.

[31] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *Science*, 286(5439):509 – 512, 1999.

[32] Cooperative association for internet data analysis. http://www.caida.org.

[33] Young Hyun. Archipelago measurement infrastructure. http://www.caida.org/projects/ark/.

[34] Young Hyun. CAIDA Monitors: The Archipelago Measurement Infrastructure. http://www.caida.org/data/monitors/monitor-map-ark.xml.

[35] Matthew Luckie. Scamper: a scalable and extensible packet prober for active measurement of the internet. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, pp. 239–245. ACM, 2010.

[36] David Meyer et al. University of oregon route views project, 2005.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California